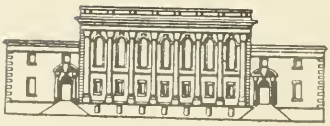
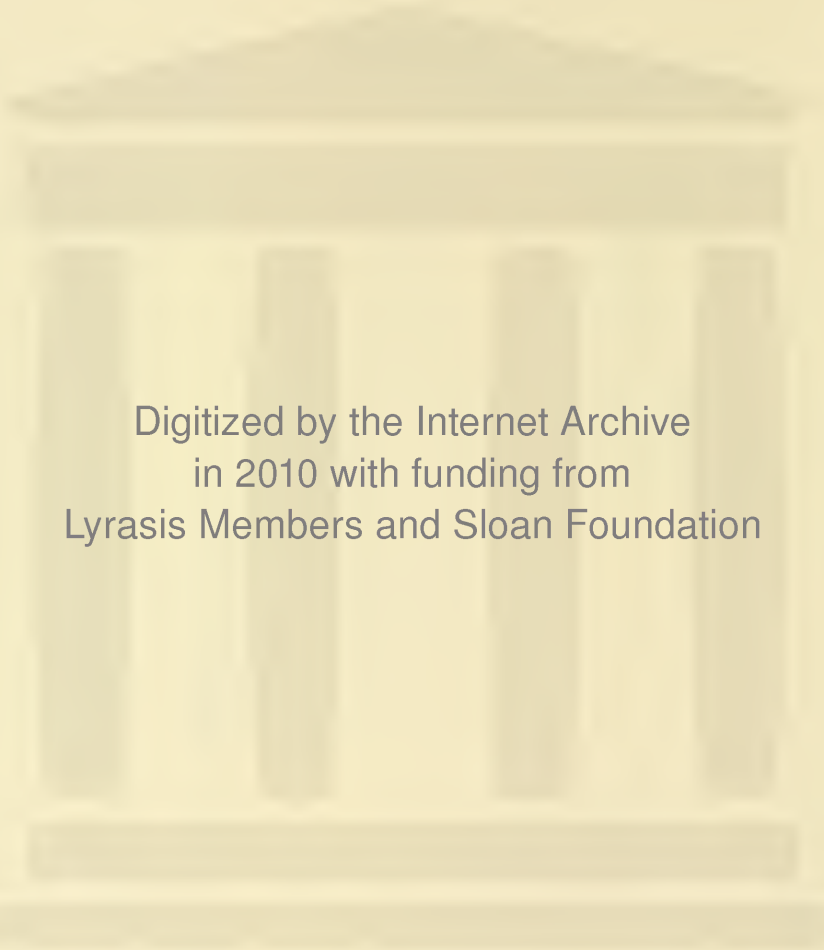
SWEET BRIAR COLLEGE
3 2449 0320006 X

THESIS
QD
461
.W55
1973

Presented by
Kathleen Eleanor Williams
Class of 1973
SWEET
BRIAR
COLLEGE
LIBRARY



SWEET BRIAR COLLEGE LIBRARY
SWEET BRIAR, VA 24595



Digitized by the Internet Archive
in 2010 with funding from
Lyrasis Members and Sloan Foundation

http://www.archive.org/details/modificationofcn00will

The Modification of a CNDO-INDO Computer Program and Its Application to the Study of the Effects of Substituents on the Electron Densities of Substituted Acetylenes and Ethylenes

by

Kathleen E. Williams

Date: May 17, 1973

Approved:

Thesis Adviser

Outside Reader

A Thesis

Submitted in Partial Fulfillment of the

Requirements for the Degree with Honors

in Chemistry

Sweet Briar College

Sweet Briar, Virginia

May, 1973

Archives
Williams
QD
461
.W55
1973

TABLE OF CONTENTS

I - INTRODUCTION.....p. 1

II- THEORETICAL.....p. 2-12

III - DISCUSSION.....p. 13-25

IV - EXPERIMENTAL.....p. 26-35

V - RESULTS AND CONCLUSIONS.....p. 36-44

Appendix A.....p.45

Appendix B.....p.46

Appendix C.....p.47

156012

I - INTRODUCTION

The purpose of this project was to gain an understanding of approximate molecular orbital theory and its uses and to apply one type of approximate molecular orbital calculation--the CNDO (complete neglect of differential overlap) method--to a series of mono- and di-substituted ethylenes and acetylenes to demonstrate the usefulness of this particular theory in studying the effects of various substituents on the σ and π densities. Particular attention was paid to the effects of the phenyl substituent on charge densities.

II - THEORETICAL

Only for two body systems is molecular orbital theory capable of providing an exact description of the electronic structure. For more complex systems--such as atoms with more than one electron or molecules involving several nuclei and their electrons--the exact solution of the Schroedinger equation becomes impossible. However, molecular orbital theory is still able to provide good approximations of many-electron, many-nuclei systems and, as such, is a useful tool for understanding molecular structure.

For a one-electron system such as the hydrogen atom, the problem reduces to one in which a solution is sought to describe the motion of the electron in the force field exerted by the nucleus. This solution is one of the wavefunctions Ψ . The Schroedinger equation for a one-electron system is

$$\left\{-\frac{h^2}{8\pi^2 m}\nabla^2 - \frac{Ze^2}{r}\right\}\Psi(1) = E\,\Psi(1) \qquad (1)$$

where h is Planck's constant, m is the electronic mass, $\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$ Z is the nuclear charge, e is the unit of electronic charge and r is the distance between the electron and the nucleus. The term inside brackets if composed of linear operators for the kinetic energy and potential energy of the system and is known as the hamiltonian operator for the system, denoted by $\hat{H}$. E is the eigenvalue of the operator[1] For a more complex system, the hamiltonian operator $\hat{H}^{total}$ is

$$\hat{H}^{total}(1, 2, \ldots, N; 1, 2, \ldots, n) = -\frac{h^2}{8\pi^2}\sum_{A}^{N} M_n^{-1}\nabla_A^2$$

(continued)

$$+\sum_{A<B} e^2 Z_A Z_B r_{AB}^{-1} - \frac{h^2}{8\pi m^2}\sum_{p}^{n}\nabla_p^2 - \sum_{A}\sum_{p} e^2 Z_A r_{Ap}^{-1} + \sum_{p<q} e^2 r_{pq}^{-1} \qquad (2)$$

where M_A is the mass of nucleus A, $Z_A e$ is the charge of A, m is again electronic mass and e electronic charge; and r is the distance between two particles i and j. Summations involving upper case letters are over nuclei, those involving lower case letters are over electrons. The Schroedinger equation for the many particle system then becomes

$$\hat{\mathcal{H}}^{\text{total}}(1,2,\ldots,N;1,2,\ldots,n)\,\Psi(1,2,\ldots,N;1,2,\ldots,n) = E\,\Psi(1,2,\ldots,N;1,2,\ldots,n) \qquad (3)$$

Now Ψ describes the motion of all particles in the molecule and E is the total energy.

As this equation (3) is insoluble in its general form, a number of approximations have been made in order to render the equation soluble to a good degree of accuracy. The Born-Oppenheimer approximation allows one to regard the nuclei as fixed points about which the electrons move. This is a reasonable approximation as the nuclei are several thousand times as massive as the electrons and move much more slowly. Mathematically, this allows the nuclear kinetic energy terms to be dropped from the hamiltonian in eq. (2). $\hat{\mathcal{H}}^{\text{total}}$ is then replaced by a purely electronic hamiltonian operator $\hat{\mathcal{H}}^{\text{el}}$.

$$\hat{\mathcal{H}}^{\text{el}} = \frac{-h^2}{8\pi m^2}\sum_{p}\nabla_p^2 - \sum_{A}\sum_{p} e^2 Z_A r_{Ap}^{-1} + \sum_{p<q} e^2 r_{pq}^{-1} \qquad (4)$$

The modified Schroedinger equation is then given by

$$\hat{\mathcal{H}}^{el}(1,2,\ldots,n)\,\Psi^{el}(1,2,\ldots,n) = \mathcal{E}\,\Psi^{el}(1,2,\ldots,n) \qquad (5)$$

where $\mathcal{E}$ is the electronic energy of the system only. The total energy E is then given by

$$E = \mathcal{E} + \sum_{A<B} e^2 Z_A Z_B r_{AB}^{-1} \qquad (6)$$

Here, the second term is the energy due to the internuclear repulsion.

The orbital approximation is also used in approximate orbital theory in attempts to obtain solutions to the many-electron Schroedinger equation. In this approximation, it is assumed that a satisfactory molecular orbital wavefunction for many-electron systems can be written as a product of one-electron functions $\psi_1, \psi_2, \ldots, \psi_n$.

$$\Psi(1,2,\ldots,n) = \psi_1(1)\,\psi_2(2)\ldots\psi_n(n) \qquad (7)$$

These one-electron functions are called orbitals and each depends on the coordinates of one electron only. The main fault with (7), however, is that it is a solution to an independent electron hamiltonian and such a hamiltonian cannot contain the interelectron repulsion term $\sum_{p<q} e^2 r_{pq}^{-1}$. In order to make some adjustment for this a modified many-electron hamiltonian operator is used which is written as a sum of "effective" one-electron hamiltonian operators $F(p)$.

$$\mathcal{F}(1,2,\ldots,n) = \sum_p F(p) = \sum_p \left[-\tfrac{1}{2}\nabla_p^2 + V(p)\right] \qquad (8)$$

$V(p)$ is the effective potential for electron p due to the potential field of the nuclei and the instananeous fields of the other electrons.

Besides the above approximations, many approximate molecular orbital theories make use of a method of obtaining solutions for many-electron systems which is based on the variational theorem.[2] This method, called the variational method, was used by Pople, Santry and Segal[3] in their CNDO (complete neglect of differential overlap) method. In its full form, applied to completely flexible functions, the variational method is equivalent to complete solution of the Schroedinger equation for the system. However, when applied to incompletely flexible functions, this method allows one to obtain approximations to the correct wavefunctions. The larger the number of adjustable parameters (i.e. the more flexible the variational function) the closer will be the calculated energies and wavefunctions to the correct values.[4] The variational method employed in most calculations is a linear combination of fixed functions.

$$\Psi(c_1, c_2, \ldots) = c_1\Phi_1 + c_2\Phi_2 + \ldots \qquad (9)$$

Here, the functions Φ_i are refered to as basis functions and are variable numerical parameters. Use of the variational method allows one to make an initial "guess" at the wavefunction and then to optimize that wavefunction by adjusting the variable parameters. One approach has been to express molecular orbitals as combinations of atomic orbitals of the constituent atoms. The simplest such approximation is the linear combination of atomic orbitals or LCAO

approach in which the molecular orbitals are expressed as simple sums of atomic orbitals with appropriate weighting coefficients.[5] A molecular orbital ψ_i may then be expressed as

$$\psi_i = c_{1i}\phi_1 + c_{2i}\phi_2 + c_{3i}\phi_3 + \ldots \qquad (10)$$

where $c_{\mu i}$ is a numerical coefficient and may be real or complex and of either sign. The LCAO approach also requires that the molecular orbitals be orthonormal.

The CNDO method makes use of the LCAO approximation to Hartree-Fock orbitals. This is often called the LCAO-Hartree-Fock self consistent field (SCF) or LCAO/SCF method. This method is used primarily with closed shell molecules and for this reason the antisymmetrized wavefunction may be written as

$$\Psi = n \sum_{P} (-1)^P P \left\{ \psi_1(1)\alpha(1)\, \psi_1(2)\beta(2) \ldots \psi_n(2n)\beta(2n) \right\} \qquad (11)$$

where P is a permutation operator, $(-1)^P$ is +1 or -1 for even or odd permutations respectively and n is a normalization constant; the ψ_i are molecular orbitals and the $\alpha(n)\beta(n)$ are spin functions. The energy expectation value $\langle\Psi|\mathcal{H}|\Psi\rangle$ is then evaluated and an expression for the Hartree-Fock electronic energy is obtained.

$$\mathcal{E} = 2\sum_{i}^{n} H_{ii} + \sum_{i}^{n}\sum_{j}^{n} (2J_{ij} - K_{ij}) \qquad (12)$$

Here H_{ii} represents the energy of an electron in molecular orbital ψ_i in the field of the bare nuclei and is given by

$$H_{ii} = \int \psi_i(1)^* H^{core} \psi_i(1)\, d\tau \tag{13}$$

where H^{core} is the one-electron hamiltonian for an electron in the field of the bare nuclei. J_{ij} is called the coulomb integral and is given by

$$J_{ij} = \iint \psi_i^*(1)\psi_j^*(2)\frac{1}{r_{12}}\psi_i(1)\psi_j(2)\, d\tau_1\, d\tau_2 \tag{14}$$

where r_{12} is the interelectronic distance. K_{ij} is the exchange integral given by

$$K_{ij} = \iint \psi_i^*(1)\psi_j^*(2)\frac{1}{r_{12}}\psi_j(1)\psi_i(2)\, d\tau_1\, d\tau_2 \tag{15}$$

Using the variational method, equations for the best molecular orbitals may be obtained by varying all the contributing one-electron functions until a minimum or stationary value for the energy is obtained.[6] This gives a value of Ψ in the form of a single determinant of spin molecular orbitals (equation (11)) which is closest to the actual value for the system. The determinants used are Slater determinants which are constructed from products of one-electron spin molecular orbitals, and are the simplest approximate orbital wavefunctions which satisfy the antisymmetry principle. The molecular orbitals obtained in this way are called self-consistent or Hartree-Fock orbitals. The wave equations take the form

$$\left[H^{core} + \sum (2J_j - K_j)\right]\psi_i = \sum_j \varepsilon_{ij}\psi_j \qquad i = 1, \ldots, n \tag{16}$$

for the orbitals $\psi_1, \psi_2, \ldots, \psi_n$. The coulomb operator J_j is given by

$$J_j(1) = \int \psi_j^*(2) \frac{1}{r_{12}} \psi_j(2)\, d\tau_2 \tag{17}$$

and the exchange operator K_j by

$$K_j(1)\psi_i(1) = \left[\int \psi_j^*(2) \frac{1}{r_{12}} \psi_i(2)\, d\tau_2\right] \psi_j(1) \tag{18}$$

Equation (18) may be written as

$$F\psi_i = \sum_j \varepsilon_{ij}\psi_j \qquad j = 1, \ldots, n \tag{19}$$

Here F is known as the Fock hamiltonian operator. With the stipulation that, since the ε_{ij} form a hermitian matrix, all $\varepsilon_{ij} = 0$ unless $i = j$ equation (19) becomes

$$F\psi_i = \varepsilon_i \psi_i \qquad i = 1, n \tag{20}$$

The eigenvalues of the Hartree-Fock hamiltonian operator are given by

$$\varepsilon_i = H_{ii}^{core} + \sum_j (2J_{ij} - K_{ij}) \tag{21}$$

Up to this point in our development of the Hartree-Fock equations ((11) through (21)) we have not made use of the LCAO approximation. Using the LCAO approach to approximate Hartree-Fock orbitals one obtains instead of equation (20)

$$\sum_{\nu} = (F_{\mu\nu} - \varepsilon_i S_{\mu\nu}) c_{\nu i} = 0 \tag{22}$$

where

$$F_{\mu\nu} = H_{\mu\nu} + \sum_{\lambda\sigma} P_{\lambda\sigma} \left[(\mu\nu|\lambda\sigma) - \tfrac{1}{2} (\mu\lambda|\nu\sigma) \right] \tag{23}$$

$$H_{\mu\nu} = \int \phi_\mu(1) H^{core} \phi_\nu(1) \, d\tau_1 \tag{24}$$

$$S_{\mu\nu} = \int \phi_\mu(1) \phi_\nu(1) \, d\tau_1 \tag{25}$$

$$P_{\lambda\sigma} = 2 \sum_{i}^{occ} c_{\lambda i}^{*} c_{\sigma i} \tag{26}$$

and

$$(\mu\nu|\lambda\sigma) = \int \phi_\mu(1) \phi_\nu(1) \frac{1}{r_{12}} \phi_\lambda(2) \phi_\sigma(2) \, d\tau_1 \, d\tau_2 \tag{27}$$

These equations for the LCAO orbitals differ from the Hartree-Fock

equations in that they are algebraic rather than differential equations. They are generally known as Roothan equations. If these equations are written in matrix form and transformed they take on the form of a standard eigenvalue problem where the elements ε_i are roots of the determinantal equation

$$\left| F_{\mu\nu}^{\tau} - \varepsilon \delta_{\mu\nu} \right| = 0 \tag{28}$$

The elements ε_i correspond to the energy eigenvalues of the molecular orbitals.

The CNDO method of Pople, Santry, and Segal[7] is the simplest approximation to (28) that still takes into account electron repulsion. In this method the inner shells of atoms are treated as a rigid core and only valence shell electrons are treated explicitly. Use is made of the zero differential overlap (ZDO) approximation

$$(\mu\nu|\lambda\sigma) = (\mu\mu|\lambda\lambda)\,\delta_{\mu\nu}\,\delta_{\lambda\sigma} \tag{29}$$

where δ_{ij} is the Kronecker delta. Also, the overlap integrals

$$S_{\mu\nu} = \int \phi_\mu(1)\,\phi_\nu(1)\,d\tau_1 \tag{30}$$

are neglected. The two-electron interaction integrals are made independent of the type of orbital and dependent only on the atom to which the orbital belongs:

$$(\mu\mu|\lambda\lambda) = \gamma_{AB} \qquad \begin{cases} \text{all } \phi_\mu \text{ on } A \\ \text{all } \phi_\nu \text{ on } B \end{cases} \tag{31}$$

The approximation is made that the interaction of $\phi_\mu \phi_\nu$ (the monatomic differential overlap) with the cores of other atoms may be neglected:

$$(\mu|V_B|\nu) = \delta_{\mu\nu} V_{AB} \tag{32}$$

where V_{AB} is the matrix representing the interaction of any valence electron on A with the core of B.

Finally, the resonance integrals

$$\beta_{\mu\nu} = (\mu|-\tfrac{1}{2}\nabla^2 - V_A - V_B|\nu) \tag{33}$$

where $-V_N$ is the potential due to the core of atom N, are assumed to be proportional to the overlap integral $S_{\mu\nu}$:

$$H_{\mu\nu} = \beta_{\mu\nu} = \beta^\circ_{AB} S_{\mu\nu} \qquad \begin{cases} \text{all } \phi_\mu \text{ on } A \\ \text{all } \phi_\nu \text{ on } B \end{cases} \tag{34}$$

where β°_{AB} are the bonding parameters and are chosen to depend only on the nature of atoms A and B. With these approximations, the Fock hamiltonian matrix elements, (23), reduce to

$$F_{\mu\mu} = U_{\mu\mu} + \left(P_{AA} - \tfrac{1}{2}P_{\mu\mu}\right)\gamma_{AA} + \sum_{B(\neq A)} \left(P_{BB}\gamma_{AB} - V_{AB}\right) \tag{35}$$

$$F_{\mu\nu} = \beta^\circ_{AB} S_{\mu\nu} - \tfrac{1}{2} P_{\mu\nu}\gamma_{AB} \qquad \mu \neq \nu \tag{36}$$

where $U_{\mu\mu}$ is the one-center term

$$U_{\mu\mu} = \left(\mu \left| -\tfrac{1}{2}\nabla^2 - V_A \right| \mu\right) \tag{37}$$

Here $P_{\mu\mu}$ and $P_{\mu\nu}$ are terms representing the orbital electron density and P_{BB} represents the electron density associated with atom B. Equation (35) is then rearranged to give

$$F_{\mu\mu} = U_{\mu\mu} + \left(P_{AA} - \tfrac{1}{2}P_{\mu\mu}\right)\gamma_{AA} + \sum_{B(\neq A)}\left[-Q_B\gamma_{AB} + \left(Z_B\gamma_{AB} - V_{AB}\right)\right] \tag{38}$$

where $Q_B = Z_B - P_{BB}$ is the net charge on atom B. The quantity $Z_B\gamma_{AB} - V_{AB}$ is the difference between the potentials due to the valence electrons and the core potential of the neutral atom B and is called the penetration integral.

The total energy of the system can be found from

$$\mathcal{E}_{total} = \tfrac{1}{2}\sum_{\mu\nu} P_{\mu\nu}\left(H_{\mu\nu} + F_{\mu\nu}\right) + \sum_{A<B} Z_A Z_B R_{AB}^{-1} \tag{39}$$

where R_{AB} is the distance between the atoms A and B.

The above are some of the basic approximations made in the general CNDO theory. For a complete discussion of the CNDO program, the reader is referred to Pople and Beveridge Approximate Molecular Orbital Theory.[8]

III - DISCUSSION

In the past decade there has been considerable work done in the development of approximate molecular orbital theories. The method upon which my research has centered is the CNDO (complete neglect of differential overlap) method developed by Pople, Santry, and Segal.[9] In a CNDO calculation, values are required for the core hamiltonian elements $U_{\mu\mu}, V_{AB}$ the overlap integrals $S_{\mu\nu}$, electron repulsion integrals γ_{AB} and the bonding parameters β°_{AB}. Two formulations of the CNDO method, CNDO/1 and CNDO/2, have been proposed which differ in the methods of approximating these values. Both are fully discussed by Pople and Beveridge in Approximate Molecular Orbital Theory.[10] CNDO/1, the earlier version of the procedure, can be used for molecules containing atoms up to and including flourine. Slater-type atomic orbitals are used for the basis set and exponents are chosen according to Slater's rules with the exception of the exponent for hydrogen for which a value of 1.2 is used. Values are determined for the overlap integral

$$S_{\mu\nu} = \int \Omega_{\mu\nu}(1)\, d\tau \qquad (1)$$

where $\Omega_{\mu\nu}$ is the charge distribution function which is a product of two Slater functions χ_{μ} and χ_{ν}. The electron repulsion integral γ_{AB} is calculated as the two-center coulomb integral for valence s electrons.

$$\gamma_{AB} = Z_B \int S_A^2(1)\,(r_{AB})^{-1}\, d\tau_1 \qquad (2)$$

V_{AB}, the interaction between a valence electron A and the core of atom B, is calculated using the valence s orbital of atom A, S_A.

$$V_{AB} = Z_B \int S_A^2(1)(r_{AB})^{-1} d\tau_1 \qquad (3)$$

where Z_B is the core charge of atom B and is taken to be a point charge at the B nucleus. The $U_{\mu\mu}$ which are the kinetic energy plus core potential terms for the atom associated with orbital ϕ_μ, are obtained from observed atomic energy levels. Bending parameters, β_{AB}° are obtained semiempirically and selected to give the best fit with accurate LCAO/SCF calculations. The CNDO LCAO/SCF equations are then solved in the following steps.[11]

1. An initial guess at the coefficients of the molecular orbitals is made by a "Huckel-type" calculation.
2. Electrons are assigned to the orbitals with the lowest energies.
3. The density matrix $P_{\mu\nu}$ is calculated and then used to form a new Fock matrix $F_{\mu\nu}$.
4. Diagonalization of $F_{\mu\nu}$ leads to a new set of coefficients.

The steps 2-4 are then repeated until the desired degree of self-consistency is reached.

It has been found that the CNDO/1 method of handling penetration integrals and the one-center atomic core integrals is not sufficient. In the CNDO/1 calculations the penetration terms give calculated bonding energies even when bond orders of two atoms are zero. This is corrected for in the CNDO/2 method by neglecting the penetration integrals. As a result, the electron core-potential integrals V_{AB} are

related to the electron repulsion integrals as follows:

$$V_{AB} = Z_B \gamma_{AB} \tag{4}$$

Although there is no satisfactory theoretical justification for the above, it does appear to compensate for errors introduced by neglecting overlap integrals.

Another way in which CNDO/2 is an improvement over CNDO/1 is in the way $U_{\mu\mu}$ is estimated. In CNDO/1 $U_{\mu\mu}$ is calculated from the ionization potential I_μ of the atomic state:

$$-I_\mu = U_{\mu\mu} + (Z_A - 1)\gamma_{AA} \tag{5}$$

Alternatively, however, atomic electron affinities A_μ can be used to determine $U_{\mu\mu}$.

$$-A_\mu = U_{\mu\mu} + Z_A \gamma_{AA} \tag{6}$$

(in both cases, the atomic orbital ϕ_μ belongs to atom A.) CNDO/2 theory makes use of both the ionization potential, as is done in CNDO/1, and atomic electron affinities. This is accomplished by using an average of equations (5) and (6).

$$-\tfrac{1}{2}(I_\mu + A_\mu) = U_{\mu\mu} + (Z_A - \tfrac{1}{2})\gamma_{AA} \tag{7}$$

Thus for CNDO/2 calculations the equations for the Fock matrix

(chapter II, eqs. (35), (36)) may be written

$$F_{\mu\mu} = -\tfrac{1}{2}(I_\mu + A_\mu) + \left[(\rho_{AA} - Z_A) - \tfrac{1}{2}(\rho_{\mu\mu} - 1)\right]\gamma_{AA} + \sum_{B(\neq A)} (\rho_{BB} - Z_B)\gamma_{AB} \tag{8}$$

$$F_{\mu\nu} = \beta^\circ_{AB} S_{\mu\nu} - \tfrac{1}{2}\rho_{\mu\nu}\gamma_{AB} \tag{9}$$

Other features of the CNDO/2 method are identical to those of the CNDO/1 except that the CNDO/2 theory has been extended to include the third-row atoms sodium through chlorine.[12] This is done by using basis sets containing 3d atomic orbitals. The sp and spd basis sets are used in the CNDO calculations in this paper. The sp set is made up of 2s and 2p functions; the spd basis set includes five 3d as well as the 3s and 3p orbitals. For these extended calculations, the bonding parameters β°_{AB} for third-row elements are estimated from values for second-row elements.

$$\beta^\circ_{AB} = \tfrac{1}{2} K \left(\beta^\circ_A + \beta^\circ_B\right) \tag{10}$$

$$\beta^\circ_A = \beta_C \frac{U_{3s,3s}(A) + U_{3p,3p}(A)}{U_{2s,2s}(C) + U_{2p,2p}(C)} \tag{11}$$

Here, K is a constant given the value 0.75 when either A or B are third-row elements and unity when they are second-row elements. The β°_C are values for the bonding parameters of second-row elements and

are used to determine the β_A° in eq. (11).

A less approximate method which is closely related to the CNDO method is the INDO (intermediate neglect of differential overlap) method of Pople, Beveridge and Dobosh.[13] The calculations of the INDO method are nearly the same as those used in CNDO/2 and the basic approximations are identical except for monatomic terms. This method involves neglect of differential overlap in all electron-interaction integrals except those involving only one center. In CNDO theory, interactions between electrons on the same atom A are replaced by γ_{AA} whether the electron spins are parallel or antiparallel. INDO theory, however, retains differential overlap in one-center integrals. For this reason, INDO is better able to accomodate unpaired spin distributions and the splitting of states arising from the same configuration. Since the basis set contains only one atomic orbital of each symmetry s, px, py, pz, off-diagonal core elements vanish and the expressions for the Fock matrix elements in the INDO method are given by

$$F_{\mu\mu}^{\alpha} = U_{\mu\mu} + \sum_{\lambda}^{A}\left[P_{\lambda\lambda}(\mu\mu|\lambda\lambda) - P_{\lambda\lambda}^{\alpha}(\mu\lambda|\mu\lambda)\right] + \sum_{B(\neq A)}(P_{BB} - Z_B)\gamma_{AB} \qquad \mu \text{ on atom } A \tag{12}$$

$$F_{\mu\nu}^{\alpha} = (2P_{\mu\nu} - P_{\mu\nu}^{\alpha})(\mu\nu|\mu\nu) - P_{\mu\nu}^{\alpha}(\mu\mu|\nu\nu) \qquad \mu \neq \nu \tag{13}$$

and

$$F_{\mu\nu}^{\alpha} = \tfrac{1}{2}(\beta_A^\circ + \beta_B^\circ)S_{\mu\nu} - P_{\mu\nu}^\circ\gamma_{AB} \qquad \begin{cases}\mu \text{ on atom } A\\ \nu \text{ on atom } B\end{cases} \tag{14}$$

The nonvanishing one-center integrals are specified using Slater's notation and assuming that the 2s and 2p orbitals have the same radial parts. For two-electron integrals these are

$$(ss|ss) = (ss|xx) = F^0 = \gamma_{AB} \tag{15}$$

$$(sx|sx) = \tfrac{1}{3} G^1 \tag{16}$$

$$(xy|xy) = \tfrac{3}{25} F^2 \tag{17}$$

$$(xx|xx) = F^0 + \tfrac{4}{25} F^2 \tag{18}$$

$$(xx|yy) = F^0 - \tfrac{2}{25} F^2 \tag{19}$$

with similar expressions for $(ss|zz)$ etc. Here, the terms F^0, G^1 and F^2 are two-electron integrals dealing with the radial parts of atomic orbitals. Slater atomic orbitals are used to evaluate the integral F^0. F^2 and G^1 are obtained semiempirically as are the values for the monatomic core integrals $U_{\mu\mu}$.

A considerable amount of work has been done on calculating equilibrium geometries and electronic charge distributions of molecules using the CNDO/2 and INDO methods. Especially useful are the calculated dipole moments, as often experimental data are available for comparison. At the level of approximation of CNDO and INDO calculations, the electric dipole moment μ is taken to be the sum of the net charges at the nuclear positions μ_{chg} and a hybridization term measuring the contribution from charge displacement away from thecenter of the nuclear position μ_{hyb}.

$$\mu = \mu_{chg} + \mu_{hyb} \tag{20}$$

Pople and Segal have done extensive calculations on diatomic molecules[14]

and Pople, Beveridge and Dobosh have done calculations on molecules of the type AB_2 and AB_3.[15] Calculated values for equilibrium bond-lengths and dipole moments for diatomic molecules agree well with values determined experimentally, while stretching force constants, binding energies and ionization potentials were less satisfactory. For AB_2 and AB_3 molecules equilibrium bond angles were well reproduced and dipole moments and binding force constants were in fairly good agreement. As expected, INDO calculated values for methyl and ethyl radicals were much better than those obtained using the CNDO method. Pople and Gordon[16] used the CNDO/2 method to calculate LCAO molecular orbitals, charge distributions and dipole moments for several organic compounds. Agreement between calculated and observed dipole moments was rather good, indicating that the calculated charge densities are reasonably accurate. Extensions of CNDO calculations to third-row atoms have been done by Santry and Segal.[17] It was shown that including 3d orbitals in the calculations causes a great deal of redistribution of electrons for third-row atoms and consequently has a large effect on the dipole moment.

Ronish and Krimm[18] have recently applied the CNDO method to the calculation of the location of side chain atoms with respect to the main chain in peptides and paraffins. Included in the calculations were the molecules propane and N-acetylalanine-N-methylamide. In both cases, the angle of the side chain with respect to the main chain was varied until the energy was minimized.

Calculations of electronic structures of neutral and protonated

adenine tautomers by the CNDO/2 and MINDO (modified intermediate neglect of differential overlap) methods have been carried out by Jordan and Sostman.[19] Relative energies of the neutral tautomers were shown to depend on the geometry assumed and the parametrization, while relative stabilities, electronic structures, dipole moments and ionization potentials were independent of small changes in the geometry. Marked reorganization of charge and increased ionization potential were noted for the protonated tautomers. The two methods of calculation gave similar dipole moments but the calculated net atomic charges were very different.

Imamura, Fujita, and Nagata[20] used the CNDO/2 molecular orbital method to calculate chemical shifts of carbon and nitrogen atoms in the DNA bases adenine, thymine, and cytosine. In these calculations, the charge redistribution effect was included. In other words, when an inner-shell electron is ionized, the valence electrons are subjected to a different core potential than that of the neutral atom. Chemical shifts of the carbon atoms correlated well with calculated binding energies and the net charges on the atoms. The chemical shifts for carbon were calculated with the CNDO/2 method without modifying the parameters included in this method, i.e. for neutral molecules, the usual parameters are used and for ionized molecules the core potential is increased by one. For nitrogen, however, better results were obtained when the CNDO/2 parameters were modified so that those of the nitrogen atom were used for the ionized carbon atom, those of oxygen for the ionized nitrogen atom, etc. Overall, agreement between

calculated and experimental shifts for nitrogen were very good. However, theoretical and observed binding energies did not show very good agreement, the calculated values being consistently much lower and of the wrong sign.

Bruns and Person[21] have used CNDO molecular orbital theory to calculate dipole moment derivatives and infared intensities of C_6H_6 and C_6F_6. Good agreement of calculated and experimentally determined values for the dipole derivatives was observed for C_6H_6 and C_6D_6. However, agreement with experimental results is much worse for C_6F_6. The effect of rehybridization of the C-H and C-F bands on the observed vibrational intensities of the two molecules C_6H_6 and C_6F_6 was mentioned, and the failure of CNDO theory to account for this was discussed.

Brownlee and Taft[22] have done calculations of charge densities for meta and para-substituted fluorobenzenes, trans-substituted fluoroethylenes, substituted fluoroacetylenes, monosubstituted benzenes, ethylene, and acetylene, and for 4-substituted[2.2.2]bicyclooctyl-1-fluorides. Special attention was given in these calculations to the effects of substituents on F orbitals. The CNDO/2 calculations were used to help interpret NMR shifts of the fluorobenzenes--with the assumption that the effect of substituents on the NMR shift may be related to the effects on π and σ charge densities of the F orbitals.

INDO calculations on planar organic free radicals have been done by Pople and Beveridge.[23] They discussed delocalization of π electrons in phenyl-substituted π radicals. It appears that when the plane of the phenyl group is coplanar with the plane of the central system, the unpaired electron is delocalized throughout the system. However, when

the phenyl group and central system are not coplanar, the delocalization is reduced. Pople and Beveridge used the INDO method to investigate this for the phenyl-substituted methyl radical. Calculations on the molecules were done as a function of the angle between the CH_2 plane and the phenyl plane. The results obtained by them showed reasonable agreement with experimentally determined values for the planar configuration except that the ortho-para values were reversed, and the meta values were too large.

Other methods of making approximate molecular orbital calculations have been developed in recent years. The MINDO method mentioned above is just one example. Baird and Dewar[24] have used this method for calculating heats of formation of hydrocarbons--for which this method is especially well adapted. MINDO apparently does not lead to good values for bondlengths nor does it give relative energies of rotational insomers.

Pople, Santry and Segal[25] have developed an invariant approximate SCF theory called NDDO (neglect of diatomic differential overlap) in which differential overlap is neglected only for atomic orbitals on different atoms. This is the next level of approximation above INDO theory. The main advantage to this type of approximation is that dipole-dipole interactions are retained. Otherwise, integral evaluation is nearly the same as for other semiempirical SCF methods. The one-center repulsion integrals for this type of calculation are the same as for INDO. Calculations comparing NDDO and <u>ab initio</u> Hartree-Fock-Roothan-SCF methods have been reported for the isomeric structures $C_2H_3^+$ and $C_2H_5^+$.[26] NDDO and the <u>ab initio</u> calculations were compared and

were found to be very close.

One method that seems to be showing much promise is the self-consistent molecular-orbital method involving Gaussian-type basis functions. In the earlier versions of these calculations, a minimal basis set of Slater type orbital (STO) basis functions is replaced by least-squares fitted linear combinations of three or four Gaussian-type orbitals (STO-3G or STO-4-G). Pople and Hehre[27] have done calculations of charge distribution with this method and found that it has many features in common with the CNDO method. They have also studied the protonation of benzene using a STO-3G minimal basis set and found that their calculated results are in good agreement with experiment.[28]

In the STO-3G and STO-4G methods, the molecular orbitals are written in the LCAO form

$$\psi_i(r) = \sum_{\mu} c_{\mu i}\, \phi_{\mu}(\zeta_{\mu}, r) \tag{21}$$

Where $\phi_{\mu}(\zeta_{\mu},r)$ is a least-squares Gaussian representation of a Slater-type orbital with Slater exponent ζ_{μ}

$$\phi_{\mu}(\zeta_{\mu},r) = \zeta_{\mu}^{3/2}\, \phi_{\mu}(1, \zeta_{\mu} r) \tag{22}$$

Equation (2) then gives for the orbitals

$$\phi_{1s}(1,r) = \sum_{k=1}^{K} d_{1s,k}\, g_{1s}(\alpha_{1k}, r) \tag{23}$$

$$\phi_{2s}(1,r) = \sum_{k=1}^{K} d_{2s,k}\, g_{1s}(\alpha_{2k}, r) \tag{24}$$

$$\phi_{2p}(1,r) = \sum_{k=1}^{K} d_{2p,k}\, g_{2p}(\alpha_{2k}, r) \tag{25}$$

where g_{1s} and g_{2p} are Gaussian-type orbitals and the constants d and α are the coefficients and Gaussian exponents in the least-squares expansion. The basis set is written as STO-KG where K is 3 or 4 denoting the number of Gaussian functions used.

Ditchfield, Hehre, and Pople[29] have presented a method of calculation using extended basis sets of atomic functions expressed as sums of four Gaussians for atoms through flourine. In this method, inner shells are represented by a basis function taken as the sum of four Gaussians and valence shells are expressed as inner and outer parts represented by three and one Gaussian functions, respectively. They have also done work using extended basis sets consisting of 5 and 6 Guassians.[30] These calculations appear to give much better values for total energies, but relative energies and equilibrium geometries are seen to have approximately the same values as those calculated using 3 and 4 Gaussians. Extended basis sets, i.e. sets containing more than the minimal number of atomic functions, have been used in calculations by Lathan, Hehre, and Pople[31] to calculate the equilibrium geometries and energies of neutral and positively charged small hydrocarbons (containing one or two carbon atoms). Ditchfield, Del Bene and Pople[32] have made ab initio calculations of excited state molecular geometries of small molecules C_2H_2, HCN, FCN,

H_2CO, FNO, F_2CO, HCF and HNO . In this case calculations were made using both a minimal and small extended basis set of Gaussian functions. Results for bond lengths and angles are reported to be good. It was noted that the extended basis sets usually gave slightly better values in cases where electronegative atoms were involved. The calculated excitation energies were in fair agreement with experimentally determined values.

From the amount of work which has been done, it is obvious that approximate molecular orbital theory has become an important tool of the chemist. Because of the great amount of calculation involved in the methods mentioned above, ways are continually being sought to improve them. Specifically, methods are sought that give more accurate values--at least for some of the calculated quantities--while requiring a minimal amount of extra time for calculations. While theories such as CNDO and INDO are adequate for many calculations, ab initio calculations involving basis sets which are combinations of Gaussian functions appear to be a very promising area of investigation.

IV - EXPERIMENTAL

The method of calculation employed in this paper is that developed by Pople, Santry and Segal,[33]--the complete neglect of differential overlap (CNDO) method. The particular version used is CNDO/2. A computer program for both CNDO and INDO calculations had previously been obtained from Dr. Paul A. Dobosh and adapted to be used on a National Cash Register Century 200 computer. The program was originally designed for an IBM 360/65 digital computer which uses a version of FULL FORTRAN IV very similar to that used by the NCR Century 200 digital computer. Compiler idiosyncrasies and the smaller memory of the Century 200 accounted for the major differences.

The original program offers the option of doing either CNDO or INDO calculations and also of working with either closed or open-shell systems. The program as it now stands has been adapted for use on the NCR Century 200 only in the sections used for CNDO calculations. In the original version, the program was capable of handling molecules with maximum sizes of 35 atoms or 80 basis functions. The smaller memory size (64K) of the NCR Century 200 forced these to be cut down to 10 atoms and 31 basis functions. If the option INDO is chosen the program is capable of doing calculations on molecules containing atoms up to and including fluorine. The CNDO option does calculations for atoms up to and including chlorine.

The MAIN part of the program inputs data and converts coordinates into atomic units. Subroutine COEFFT stores the values of coefficients used in calculating the coulomb and overlap integrals which is done in subroutine INTGRL. INTGRL calls RELVEC, HARMTR, MATOUT and SS which in

turn calls subroutines AINTGS and BINTGS. Subroutines HUCKCL, SCFCLO and CPRINT (or, for an open-shell molecule, HUCKOP, SCFOPN and OPRINT) are then called by MAIN to do the MO (molecular orbital) calculation. These subroutines call other subroutines--EIGN, SCFOUT and EIGOUT--during calculations. RELVEC calculates the vector distance between pairs of basis functions. Each of these pairs then forms an element of the overlap matrix. This matrix is N×N where N is the number of basis functions in the molecule. SS computes the reduced overlap integrals which are then transformed by the rotation matrix calculated in HARMTR to give the overlap integral $S_{\mu\nu}$. The one and two-center coulomb integrals are also computed by using the reduced overlap integrals.

HUCKCL does a Huckel-type approximation to the Fock matrix. This matrix is then diagonalized and an initial density matrix is formed. At this point corrections are made to the core hamiltonian matrix.

In SCFCLO the initial density matrix and core hamiltonian are used to calculate the Fock matrix. This is then diagonalized by EIGN, a new density matrix is constructed and used to form a new Fock matrix. The procedure is repeated until the calculated electronic energy converges to within the desired limits. Then the Fock matrix is printed, diagonalized again and used to calculate the eigenvectors.

In CPRINT, electron densities, dipole moments and nuclear repulsion energies are calculated. SCFOUT and EIGOUT print out matrices and eigenvalues respectively.

The corresponding subroutines HUCKOP, SCFOPN and OPRINT are

similar to HUCKCL, SCFCLO and CPRINT except that the calculations must take into account open-shell electronic configuration.[34]

Other modifications in the original program that had already been made by a previous user of the program, Katherine Upchurch,[35] included the replacing of the IMPLICIT REAL * 8 statement with the DOUBLE PRECISION statement as the former is not available in NCR FULL FORTRAN IV. The DOUBLE PRECISION statement also had to be declared for each real variable as the NCR language contains no provision for automatically declaring every real variable double precision. The file numbers on the READ and WRITE statements were changed to 1 and 3 respectively and the DFLOAT statement was removed from the program (the DFLOAT statement, which converts integer variables to floating decimal point variables in mixed-mode arithmetic is not needed in NCR FORTRAN IV as conversion to floating point variables is done automatically).

In order to save memory space, the COEFFT subroutine was removed from the program and the Y and Z coefficients used in calculating the reduced overlap integrals were fed in as data. This was done because many of the Y and Z coefficients stored in memory in COMMON/ARRAYS/ were zero. The Y and Z values were also declared double precision, which meant that much of the memory space taken up by COEFFT in the program held zero values. Reading the Y and Z coefficients as data probably takes more computer time and intailed considerable changes in parts of the program dealing with the determination of appropriate Y and Z coefficients. However, much free memory space was obtained in this way.

The program has also been modified somewhat since then by this user. (a copy of the program as it now stands is found in appendix A along with an example of the output from calculations on a simple molecule.)

As it was desirable to have the program put on disk to eliminate the need for the large deck of cards, this was done. A provision for making a tape of the Y and Z coefficients and then reading the values off of the tape was also added. This was done in order to eliminate the need for including the Y and Z coefficients in the data deck and also made it easy to remake the tape in the event that it was accidentally destroyed. This was accomplished by the inclusion of statements in the MAIN section instructing the computer to write the values on tape and then to read them off of the tape. The option MTAPE was added to the statement giving the wavefunction option and open or closed shell option.

```
READ (1,40) OPTION, OPNCLO, MTAPE                    (1)
```

The options are 1 for making the tape and 0 otherwise. For example, if the user wishes to make a closed-shell, CNDO calculation and to have the tape of the Y and Z coefficients made, the input would look like:

```
CNDO CLSD 1                                          (2)
```

The computer would then be instructed to read the Y and Z values and write them on tape to be read later in the program. If MTAPE equals 0, these steps are skipped and the values are read off of an already available tape.

Since each run took so long on the computer (approximately 45 minutes to an hour for each molecule--even rather simple ones) a way was sought to allow the technicians at the Educational Computer Center to run one molecule out of the data deck at a time. In order to do this, a DATA statement was added

```
DATA MORE/4HMORE/,LAST/4HLAST/                       (3)
```

and the NRUN statement (which input the number of runs to be done) was changed. Instead of inputing the number of runs, a MORE or LAST card (FORMAT (A4)) was inserted behind the data cards of each molecule. A statement, READ (1, 12) NRUN, was inserted in MAIN and the statement

IF (NRUN. EQ. MORE) GO TO 11 (4)

was inserted after the CALL statements in MAIN. If, after doing calculations on the first molecule in the data stack, a MORE card is encountered, the computer returns to statement 11 and begins calculations on the next molecule. If a LAST card is read, however, the program terminates.

A "counter" KRUN was also added so that once the and values had been read off the tape (whether the tape had just been made or was already available) and were stored in memory, no attempt would be made to read them off of the tape a second time during the same series of calculations. This was accomplished by initially setting KRUN 0 and then adding an increment of 1 each time calculations on a molecule were done. A statement instructing the computer to ignore the instructions for making the tape after the first run is completed is included.

IF (KRUN.GE.2) GO TO 250 (5)

In this case, the first time through the calculations, KRUN will acquire the value 1, and provided MTAPE=1, a tape will be made and the Y and Z values read from it. When calculations begin on the second molecule, KRUN will take on the value 2 and since in (5) if KRUN is greater than or equal to 2 the computer is instructed to go to statement 250 and continue from there, the instructions for making the tape will be omitted. The major changes in MAIN in the order in which they occur are given below; statements marked with an asterick are those which were added or changed in some way. The

WRITE (9) statements instruct the computer to write on tape, READ (8) means read off tape.

```
     *  DATA MORE/4HMORE/,LAST/4HLAST/                          (6)
     *  KRUN  0                                                 (7)
11      CONTINUE                                                (8)
        READ (1,20) (AN(I), I = 1,10)                           (9)
        WRITE (3,30) (AN(I), I = 1,10)                          (10)
     *  READ (1,40) OPTION, OPNCLO, MTAPE                       (11)
        WRITE (3,45) OPTION, OPNCLO                             (12)
40   *  FORMAT (A4,1X,A4,I2)                                    (13)
     *  KRUN KRUN 1                                             (14)
     *  IF (MTAPE.EQ.0) GO TO 200                               (15)
        DO 7 I 1,87                                             (16)
7    *  READ (1,75) NZY(I),NZZV(I)                              (17)
        DO 8 I 1,224                                            (18)
8    *  READ (1,85) NZY(I),NZYV(I)                              (19)
     *  WRITE (9) (NZZ(I), I=1,87),(NZZV(I),I=1,87)             (20)
     *  WRITE (9) (NZY(I), I=1,224),(NZYV(I),I=1,224)           (21)
     * REWIND 9                                                 (22)
200  *  IF(KRUN.GE.2) GO TO 250                                 (23)
     *  IF (MTAPE. EQ.1) GO TO 250                              (24)
     *  READ (8) (NZZ(I),I=1,87),(NZZV(I),I=1,87)               (25)
     *  READ (8) (NZY(I),I=1,224),(NZYV(I),I=1,224)             (26)
     *  REWIND (8)                                              (27)
          .
          .
          .
          .
10      CONTINUE                                                (28)
```

```
      *  READ (1,12) NRUN                                    (29)
12       FORMAT (A4)                                         (30)
           .
           .
           .
           .
         CALL CPRINT                                         (31)
      *  IF (NRUN.EQ.MORE) GO TO 11                          (32)
```

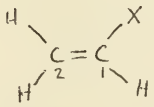
Once the above changes had been made, several runs were tried with the program running off disk and the Y and Z coefficients on tape. Unfortunately, when the runs were checked against previous runs of the same molecule, it was found that except for the first molecule in each run, the calculated values were faulty. At first it was thought that a long continuation statement in subroutine INTGRL might be the source of the problem, as there was a possibility that the new compiler at the computer center might not accept a continuation statement of that length as the old compiler had. However, breaking the statement up into smaller statements did not help. Some time was spent on this problem and the tentative conclusion was reached that the faulty calculations were somehow due to a failure of the computer to retain the correct values for the Y and Z coefficients after the first molecule had been run. At this time there is no explanation as to why this should occur. In order to get around the problem, however, each molecule has been run separately. Except for the length of time required for each fun, no other major problems have been encountered.

As the NCR Century 200 digital computer has only a limited amount of memory space (64K) it was not possible to run calculations of

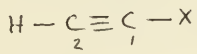
molecules larger than 10 atoms. However, several runs of various phenyl substituted ethylenes and acetylenes were done on the CDC 6400 computer at the University of Virginia. For these runs the full 35 atom-80 basis function version of the program was used, as the memory space available in the CDC 6400 was more than adequate.

The molecules studied were substituted ethylenes and acetylenes with emphasis on flourine, chlorine and phenyl substituents. The substituents considered were F, Cl, CN, CHO, NO_2 and phenyl. Standard angles and bondlengths were employed for all molecular geometries as suggested by Pople and Beveridge.[36] Angles for all double bonded atoms were assumed to be trigonal--i.e. 120°, and all triply-bonded carbon atoms were assumed to have bond angles of 180°. Bond lengths used are given in Appendix B.

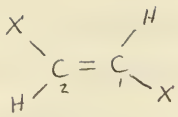
The use of standard models for molecular geometries in this case is considered valid since qualitative rather than quantitative information is sought from calculations at the level of the CNDO approximation. The configurations of the various types of molecules considered are given in Fig. 1



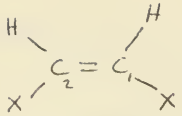
mono-substituted ethylene



mono-substituted acetylene



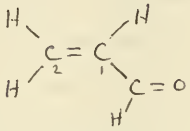
trans di-substituted ethylene



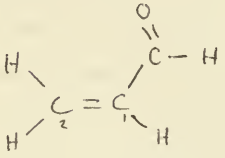
cis di-substituted ethylene

Fig. 1

For molecules containing the CHO group, the configurations are those given in Fig. 2.



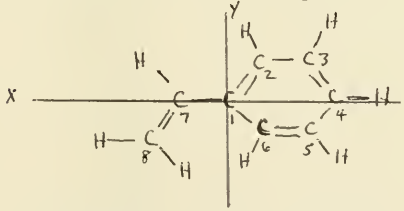
"trans"



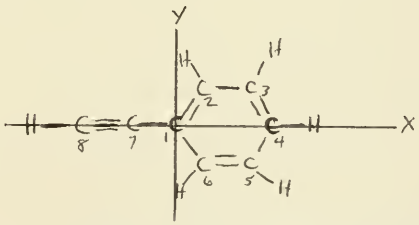
"cis"

Fig. 2

For the cartesian coordinates of all of these configurations, the molecules were considered to lie in the X-Y plane with C_1 at the origin. In the case of phenyl substituted molecules, however, the coordinate axes were placed as shown in Fig. 3, again assuming that the molecule lies entirely in the X-Y plane.



phenyl-substituted ethylenes



phenyl-substituted acetylenes

Fig. 3

Examples of the coordinates used are given in Appendix C.

In the calculations done on molecules with F, Cl, CHO, CN and NO_2 substituents the criterion for convergence was a change in the calculated electronic energy of less than 10^{-4} Hartree between successive calculated values. The calculations done with phenyl as a substituent were done using the expanded-memory version of the program. In some instances, there was severe oscillation between successive values for the electronic energy and this criterion was never met. This most often seemed the case for di-substituted molecules containing F or CN as one or both of the substituents.

V - RESULTS AND CONCLUSIONS

By comparing electron densities on the two carbon atoms C_1 and C_2 of the substituted ethylenes and acetylenes, it is possible to make some qualitative statements about relative electronegativities and inductive effects. In all the calculations, it is assumed that addition of substituents to the molecule does not change the bond lengths or bond angles. This is justifiable considering the comparative approach being used. Also, experimental data for comparison was not always available for the molecules considered. Since the convention of placing the double or triple bond along the X axis with the Z axis perpendicular to the plane of the molecule was used, π electron densities could be obtained directly from the density matrix. The σ electron density is calculated as the sum of the contributions from the diagonal S, Px and Py elements of the charge density matrix.

The two halogens used as substituents, F and Cl, should cause quite a difference in relative electron densities on C_1 and C_2 in the corresponding molecules. Fluorine is the smaller atom and the most electronegative of all the elements. Its inductive effects are greater than Cl meaning that it tends to draw σ electrons to itself more readily than Cl. It also has a larger reasonance effect--it tends to delocalize π electrons to a greater degree that Cl. The smaller fluorine atom has a much higher charge density than chlorine. Because of this, there is considerable repulsion between electron pairs. The formation of π bonds lessens the charge density on the atom. Fluorine's tendency to contribute to π bonds is, therefore, greater than that for chlorine.

One expects then, to see a greater deficit in σ electron population of a carbon atom bonded to F than one bonded to Cl. This is born out in the CNDO calculations as can be seen in tables 1-3.

TABLE 1

(Note: electron densities are given as the difference between the calculated valence electron population of the atom and the valence electron population of the neutral atom. A + indicates a gain, - indicates a loss.)

electron densities of mono-substituted ethylenes

molecule	σ		π	
C_2 C_1	C_1	C_2	C_1	C_2
$H_2C=CH_2$ *	+.03	+.03	+.00	+.00
$H_2C=CHF$ *	-.17	+.05	-.01	+.04
$H_2C=CHCl$ *	-.05	+.015	-.02	+.04
$H_2C=CH-CHO$ "cis"*	+.03	+.05	+.02	-.06
$H_2C=CH-CHO$ "trans"*	+.03	+.05	+.01	-.05
$H_2C=CH-NO_2$ *	-.14	+.004	-.01	+.11

TABLE 2

electron densities of di-substituted ethylenes

Molecule	σ		π	
C_2 C_1	C_1	C_2	C_1	C_2
$FHC=CHCl$	-.28	-.39	+.38	+.45
$FHC=CH-CN$ *	-.01	-.20	+.08	-.05
$FHC=CH-NO_2$ *	-.14	-.21	+.06	-.08
$FHC=CH-CHO$ cis*	+.03	-.17	+.09	-.09
$FHC=CH-CHO$ trans *	+.03	-.16	+.08	-.07
$ClHC=CH-NO_2$*	-.06	+.01	+.05	-.08
$ClHc=Ch-CHO$ trans *	-.09	-.61	-.19	-.52

* See reference 37

TABLE 3

electron densities of substituted acetylenes

molecule	σ		π	
C_2 C_1	C_1	C_2	C_1	C_2
HC≡CF	-.09	+.10	-.02	+.06
HC≡CCl	-.02	+.06	-.01	-.02
HC≡C-CHO	+.01	+.10	+.02	-.05
HC≡C-NO_2	-.075	+.09	+.05	-.08
FC≡C-CN	+.02	-.19	+.06	-.03
FC≡CF	-.16	-.16	+.04	+.04
ClC≡CCl	-.02	-.02	-.02	-.02
ClC≡CCN	-.03	-.04	+.04	-.03
ClC≡C-NO_2	- .08	+.01	+.05	-.08

For example, the σ electron deficit at C_2 in ClHC≡$CHNO_2$ is much less than at C_2 in FHC≡$CHNO_2$. The corresponding σ density at C_1 in the fluoro-substituted molecule is less than for C_1 in the chloro-substituted one--showing that the inductive effects of Fluorine extend to this part of the molecule also.

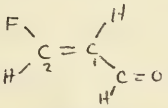
From tables 1 and 2, it can be seen that the calculated order of inductance strength of the substituents is:

$$F > NO_2 > Cl > CN > CHO \simeq H \qquad (1)$$

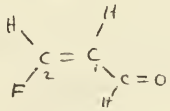
the CHO and H apparently being very close. This ordering correlates very well with observed relative inductive strength.[38]

Table 2, though not complete, gives us a chance to compare the inductive strengths of the substituents on the same molecule. It is interesting to note the competition for σ electrons of F and NO_2 for instance. Both are very inductive but now it can be seen more clearly than in table 1 that F is much stronger in this respect than NO_2. Comparison of the values for C_1 and C_2 of $FHC=CHCl$ shows again how much stronger F is than Cl in this respect. The fact that the C_2 value is actually slightly positive for $ClHC=CH\text{-}NO_2$ points out just how strong the inductive effects of NO_2 are and supports the ordering given in (1).

It is also interesting to note the slight differences in σ electron density on the respective carbon atoms of the cis and trans forms of $FHC=CH\text{-}CHO$. Although very close to each other in relative electron densities, the trans formation seems to have a slightly higher σ electron density on carbon 1 than the cis formation. The two molecules look like:



trans



cis

The data in table 3 shows the same trends as that in tables 1 and 2. It is quite evident here how much stronger the inductive effects of fluorine are than of chlorine. In difluoro acetylene, the σ electron deficit is -.16 on each carbon as compared with only -.02 on each of the carbons in the dichloro compound.

The π densities given in table 1 also show a certain ordering.

$$Cl > NO_2 \simeq F > H > CN > CHO\ trans \simeq CHO\ cis \qquad (2)$$

One notices immediately that chlorine and fluorine have changed places in the ordering--it seems that the chlorine draws π electrons to it more readily than fluorine and that the fluorine is making a greater contribution to π bonding than the chlorine. The total π density at the carbons attached to chlorine is less than the total π density at comparable carbon atoms attached to fluorine. This is what one would expect.[39] It is also interesting to note that the π density of the carbon bonded to the fluorine or chlorine atom is considerably less than the π density at C_2. This is not what one might expect, as it would seem that the F and Cl atoms would cause the π density at C_1 to be greater. However, this could be due to a kind of alternation of charge in the molecule. The data on cis and trans CHO substituted ethylenes-where an additional carbon atom is involved-seems to bear this out. Here the π density is indeed greater on C_1 than C_2.

The phenyl-substituted ethylenes and acetylenes show some interesting characteristics. (see table 4) Alternation of π bond density around the ring is quite evident in $H_2C{=}CH\text{-}C_6H_5$. The C_1, C_3, C_5, and C_7, positions are all slightly deficient of electrons while C_2, C_4, C_6, and C_8, have a slight excess of electrons. C_2, C_4, and C_6correspond to the ortho and para positions on the benzene ring. The increased π electron density at these postitons indicates that the sites are more vulnerable to attack in an electrophilic aromatic substitution reaction. This is indeed true and C_2H_3 is a very weakly activating ortho-para director.[40]

In the cis and trans fluoro- and chloro-substituted phenyl ethylenes C_7 becomes more deficient in electrons--as would be expected.

Also the π density on C_8 increases--which would be due to the attachment of the fluorine and chlorine to that atom. Charge densities within the ring itself, however, do not change as much when fluorine is the substituent as when chlorine is. Apparently, the Cl atom exerts more of an electron withdrawing effect on the ring than F. When chlorine is the substituent, C_4, which has an excess of electron density in

TABLE 4

π densities of phenyl substituted ethylenes and acetylenes.

	$H_2C{=}CH\text{-}C_6H_5$	$FH\text{-}C{=}CH\text{-}C_6H_5$ cis	trans
C_1	-.01	-.01	-.02
C_2	+.01	+.02	-.02
C_3	-.01	-.01	-.01
C_4	+.005	+.01	+.01
C_5	-.01	-.01	-.01
C_6	+.01	+.008	+.02
C_7	-.02	+.05	-.05
C_8	+.02	+.00	+.01

	$ClH\text{-}C{=}CH\text{-}C_6H_5$ cis	trans
C_1	-.01	+.00
C_2	+.007	+.004
C_3	-.01	-.01
C_4	+.00	-.004
C_5	-.01	-.01
C_6	+.009	+.007
C_7	-.04	-.05
C_8	+.03	+.03

	$HC\equiv C\text{-}C_6H_5$	$Cl\text{-}C\equiv C\text{-}C_6H_5$	$F\text{-}C\equiv C\text{-}C_6H_5$
C_1	-.01	+.00	-.01
C_2	+.01	+.004	+.01
C_3	-.006	-.01	-.01
C_4	+.00	-.006	+.00
C_5	-.006	-.01	-.01
C_6	+.01	+.004	+.01
C_7	+.00	-.01	+.06
C_8	+.00	-.004	-.02

Note: Values for which three figures are given are uncertain in the third decimal place.

the other phenyl substituted ethylenes, develops a deficit of π electrons. Both Fluorine and chlorine have a correspondingly greater affect on the C_2, C_4, and C_6 positions than the C_1, C_3 or C_5 positions. This is interesting, as fluorine and chlorine are both deactivating (electron withdrawing) ortho-para directors when attached directly to the benzene ring itself. Apparently they continue to exert this effect even when separated from the ring by the ethylene part of the molecule. This again is what would be expected, as the C=C appears to act as part of the ring.

For the phenyl substituted acetylenes much the same seems to apply. However, the C_7 position in phenyl acetylene does not have a deficit of π electrons as the phenyl substituted ethylenes did. When flourine is attached to the C_8 position the electron population of C_7 increases; however, when chlorine is attached there the π electron population becomes negative. From this it would seem that the Cl atom has a much stronger withdrawing effect on that position. Again C_4 also becomes negative for the chloro-substituted but not for the fluoro-substituted molecule. On both molecules, C_8 loses π electron population to the strongly electronegative atom attached to it.

For the most part, CNDO theory seems to work fairly well to describe electron distribution in small organic molecules. There are, however, difficulties with this method. The standard geometries employed are probably not as accurate to describe the systems as one would like. Again, it must be emphasized that the data obtained from such calculations is useful only for qualitative rather than quantitative

statements.

The problem of oscillation in the electronic energy calculations for some molecules also needs to be overcome. If the program were modified to use electronic density as the criterion for achieving self-consistency rather than the electronic energy criterion now employed, convergence might be obtained with less difficulty. This could be done by requiring the root-mean-square change in the density matrix to be less than a specified value.

In general, however, the CNDO method does seem capable of reproducing many trends and appears to be a useful tool in studying closed-shell molecular systems.

APPENDIX A

Appendix A is a copy of the CNDO program used for the calculations in this paper, and a sample run of a molecule - ClC≡CH

The Format for input of data on cards is also given below.

Format for Input

1st card - identification-i.e.name or formula of molecule

2nd card - wavefunction option (CNDO or INDO), open or closed shell option (OPEN or CLSD) and option for making a tape of the Y-Z coordinates (1 to make tape, 0 to not make the tape) These are typed as follows: 4 letters, space, 4 letters, space, 1 or 0. (A4,1X,A4,I2)

3rd card - Number of atoms, charge, multiplicity. These are typed, right-justified, in 3 blocks of 4 spaces each. (3I4)

Cards containing the atomic number and X, Y and Z coordinates of each atom in the molecule - one card for each atom. The format for these is: atomic number - right justified in the first 4 spaces. X coordinate, Y coordinate, Z coordinate - in 3 blocks, each containing 3 spaces then a field of 12 spaces for digits with the decimal point after the 7th digit. (I4,3(3X,F12.7)).

Last card - Type MORE if another molecule is to be run, LAST if not.

```
N IV  RF04     PROGRAM CNINDO  00                SOURCE      1

NO.      171429


    LABEL CONT  ...1..../....2..../....3..../....4..../....5..../....6..../....7.
    00000  0    PCNINDO    FORT/FN          EN1   N  NN      281 064   S2WILLIAMS
    *            P          AUTHORWILLIAMS KATHY
    *            F        01001 P1108001                        S
    *            F        03622 P01136              060
    *            F        09145 0 1024  1024  SCR      CNOOZY                 M01R
    *            F        08145 S 1024  1024  SCR       CNOOZY                M01R
    *            F          24                        S
    C       PROGRAM CNINDO
                 DOUBLE PRECISION ABC, C, XYZ, A, B
                 COMMON/ARRAYS/ABC(1922),NZZ(87),NZZV(87),NZY(224),NZYV(224)
                 COMMON/PERTBL/EL(18)
                 COMMON/INFO/NATOMS,CHARGE,MULTIP,AN(10),C(10,3),N
                 COMMON/ORB/ORB(9)
                 COMMON/GAB/XYZ(630)
                 COMMON/INFO1/CZ(10),U(31),ULIM(10),LLIM(10),NELECS,OCCA,OCCB
                 COMMON/OPTION/OPTION,OPNCLO,HUCKEL,CNDO,INDO,CLOSED,OPEN
                 COMMON/AUXINT/A(17),B(17)
                 INTEGER  OPTION,OPNCLO,HUCKEL,CNDO,INDO,CLOSED,OPEN
                 INTEGER ORB,EL,AN,CHARGE,CZ,U,ULIM,OCCA,OCCB
    C       INPUT IS READ IN THE FOLLOWING ORDER
    C   (1)AN IDENTIFICATION CARD WHICH IS PRINTED AT THE BEGINNING OF THE RU
    C   (2)OPTION(WAVE FUNCTION OPTION) AND OPNCLO(OPEN OR CLOSED SHELL)
    C           THE FORMAT IS A4,1X,A4  AND THE KEY WORDS ARE=
    C             FOR THE WAVEFUNCTION(A4)     CNDO     INDO
    C             FOR THE OPEN=CLOSED OPTION(A6)     OPEN    CLSD
    C   (3)NATOMS,CHARGE,MULTIP    FORMAT(3I4)
    C   (4)ATOMIC NUMBER, X COORDINATE, Y COORDINATE, Z COORDINATE - 1 CARD/A
    C         FORMAT(I4,3(3X,F12.7))
                 DATA MORE/4HMORE/,LAST/4HLAST/
                 KRUN=0
       11        CONTINUE
                 READ (1,20) (AN(I),I=1,10)
                 WRITE (3,30) (AN(I),I=1,10)
                 READ (1,40) OPTION,OPNCLO,MTAPE
                 WRITE (3,45)OPTION,OPNCLO
       40        FORMAT(A4,1X,A4,I2)
                 KRUN=KRUN+1
                 IF (MTAPE.EQ.0) GO TO 200
                 DO 7 I=1,87
        7        READ (1,75) NZZ(I),NZZV(I)
                 DO 8 I=1,224
        8        READ (1,85) NZY(I),NZYV(I)
                 WRITE(9) (NZZ(I),I=1,87),(NZZV(I),I=1,87)
                 WRITE(9) (NZY(I),I=1,224),(NZYV(I),I=1,224)
                 REWIND 9
      200        IF (KRUN.GE.2) GO TO 250
                 IF (MTAPE.EQ.1) GO TO 250
                 READ(8) (NZZ(I),I=1,87),(NZZV(I),I=1,87)
                 READ (8) (NZY(I),I=1,224),(NZYV(I),I=1,224)
                 REWIND 8
      250        CONTINUE
```

```
IV  RF04     PROGRAM CNINDO  00                SOURCE      1                    DA

   LABEL CONT  ...1..../....2..../....3..../....4..../....5..../....6..../....7..
               READ(1,50) NATOMS,CHARGE,MULTIP
               WRITE(3,60) NATOMS,CHARGE,MULTIP
               DO 10 I = 1,NATOMS
               READ(1,70) AN(I),C(I,1),C(I,2),C(I,3)
               WRITE(3,70) AN(I),C(I,1),C(I,2),C(I,3)
     C      CONVERSION OF COORDINATES FROM ANGSTROMS TO ATOMIC UNITS
               DO 9 J=1,3
          9    C(I,J) = C(I,J)/.52916700
         10    CONTINUE
               READ(1,12) NRUN
         12    FORMAT(A4)
               IF (OPTION.EQ.CNDO) GO TO 6
          1    DO 5 I=1,NATOMS
               IF (AN(I).LE.9) GO TO 4
          2    WRITE(3,3)
          3    FORMAT(5X,46HTHIS PROGRAM DOES NOT DO INDO CALCULATIONS FOR,
              1    51H MOLECULES CONTAINING ELEMENTS HIGHER THAN FLUORINE)
               STOP
          4    CONTINUE
          5    CONTINUE
          6    CONTINUE
               CALL INTGRL
               CALL HUCKEL
               CALL SCFCLD
               CALL CPRINT
               IF (NRUN.EQ.MORE) GO TO 11
         20    FORMAT (10A4)
         30    FORMAT(1H1,5X,20A4)
         45    FORMAT(5X,A4,1X,A4)
         50    FORMAT(3I4)
         60    FORMAT(/5X,I4,18H ATOMS   CHARGE =,I4,18H   MULTIPLICITY =,I4/)
         70    FORMAT(I4,3(3X,F12.7))
         75    FORMAT (8X,I3,2X,I6)
         85    FORMAT (8X,I4,2X,I6)
               CALL EXIT
               STOP
               END

' ALLOCATION

      03C10 NZZ        03D6C NZZV       03EC8 NZY        04248 NZYV

' ALLOCATION

' ALLOCATION

MS     00004 CHARGE     00008 MULTIP     0000C AN         00034 C
```

```
N IV  RF04     PROGRAM CNINDD  00                SOURCE      1

    LABEL CONT  ...1..../....2..../....3..../....4..../....5..../....6..../....7
 ALLOCATION


 ALLOCATION


 ALLOCATION

       00028 U           000A4 ULIM        000CC LLIM        000F4 NELECS
       000FC OCCB

 ALLOCATION

N      00004 OPNCLD      00008 HUCKEL      0000C CNDD        00010 INDD
D      00018 OPEN

 ALLOCATION

       00088 B

ROUTINE

QIOEI       QFINIO      QFWRIT      QUWRIT      QREWND      QDEFIN
QIOED       QSTOP       INTGRE      HUCKCL      SCFCLO      CPRINT

OCATION

       0006C LAST        00070 KRUN        00074 I           00078 MTAPE
       00080 NRUN

CATION
```

```
A IV  RF04     PROGRAM CNINDO  00              SOURCE      1                    DA

     LABEL CONT  ...1....+....2....+....3....+....4....+....5....+....6....+....7..

                 BLOCK DATA
                 COMMON/ORB/ORB(9)
                 COMMON/PERTBL/EL(18)
                 COMMON/OPTION/OPTION,OPNCLO,HUCKEL,CNDO,INDO,CLOSED,OPEN
                 INTEGER OPTION,OPNCLO,HUCKEL,CNDO,INDO,CLOSED,OPEN
                 INTEGER ORB,EL
                 DATA CNDO/4HCNDO/
                 DATA INDO/4HINDO/
                 DATA OPEN /4HOPEN/
                 DATA CLOSED/4HCLSD/
                 DATA ORB(1)/4H    S/
                 DATA ORB(2)/4H   PX/
                 DATA ORB(3)/4H   PY/
                 DATA ORB(4)/4H   PZ/
                 DATA ORB(5)/4H  DZ2/
                 DATA ORB(6)/4H  DXZ/
                 DATA ORB(7)/4H  DYZ/
                 DATA ORB(8)/4HDX2-Y2/
                 DATA ORB(9)/4H  DXY/
                 DATA EL(1)/4H    H/
                 DATA EL(2)/4H   HE/
                 DATA EL(3)/4H   LI/
                 DATA EL(4)/4H   BE/
                 DATA EL(5)/4H    B/
                 DATA EL(6)/4H    C/
                 DATA EL(7)/4H    N/
                 DATA EL(8)/4H    O/
                 DATA EL(9)/4H    F/
                 DATA EL(10)/4H   NE/
                 DATA EL(11)/4H   NA/
                 DATA EL(12)/4H   MG/
                 DATA EL(13)/4H   AL/
                 DATA EL(14)/4H   SI/
                 DATA EL(15)/4H    P/
                 DATA EL(16)/4H    X/
                 DATA EL(17)/4H   CL/
                 DATA EL(18)/4H   AR/
                 END

 ALLOCATION

 ALLOCATION

 ALLOCATION

ON      00004 OPNCLO      00008 HUCKEL      0000C CNDO        00010 INDO
ED      00018 OPEN
```

```
IV  RF04     PROGRAM CNINDO  DO                SOURCE      1                    DA

   LABEL CONT  ...1..../....2..../....3..../....4..../....5..../....6..../....7..

               SUBROUTINE INTGRL
               DOUBLE PRECISION S, XX, C, XXX, GAMMA, P, T, TEMP, C1, C2, YYY,
              1  A, B, MU, NUM, K1, K2, NEWS, NEWV, PAIRS, TERM1, TERM2, DEN
     C      ATOMIC INTEGRALS FOR CNDO CALCULATIONS
               COMMON/ARRAYS/S(31,31),XX(961),DUMMY(622)
               COMMON/INFO/NATOMS,CHARGE,MULTIP,AN(10),C(10,3),N
               COMMON/INFO1/CZ(10),U(31),ULIM(10),LLIM(10),NELECS,IOCCA,IOCCB
               COMMON/GAB/XXX(155),GAMMA(10,10),T(9,9),PAIRS(9,9),TEMP(9,9)
              1   ,C1(3),C2(3),YYY(126)
               COMMON/AUXINT/A(17),B(17)
               COMMON/OPTION/OPTION,OPNCLO,HUCKEL,CNDO,INDO,CLOSED,OPEN
               DIMENSION MU(18),NC(18),LC(9),MC(9),E(3)
               DIMENSION P(31,31)
               EQUIVALENCE (P(1),XX(1))
               INTEGER AN,ULIM,LLK,ULL,CZ,U,CHARGE,ANL,ANK,IOCCA,IOCCB
               INTEGER  OPTION,OPNCLO,HUCKEL,CNDO,INDO,CLOSED,OPEN
     C      DETERMINATION OF SIZE OF AO BASIS IN AND CORE CHARGE CZ
               N=0
               DO 60 I=1,NATOMS
               LLIM(I) = N+1
               K=1
               IF (AN(I).LT.11) GO TO 20
        10     N=N+9
               CZ(I)=AN(I)-10
               GO TO 50
        20     IF (AN(I).LT.3) GO TO 40
        30     N=N+4
               CZ(I) = AN(I)-2
               GO TO 50
        40     N=N+1
               CZ(I)= AN(I)
        50     CONTINUE
               ULIM(I) = N
        60     CONTINUE
     C      FILL U ARRAY---U(J) IDENTIFIES THE ATOM TO WHICH ORBITAL J IS
     C      ATTACHED E.G. ORBITAL 32 ATTACHED TO  ATOM 7, ETC.
               DO 70 K=1,NATOMS
               LLK = LLIM(K)
               ULK = ULIM(K)
               LIM = ULK+1-LLK
               DO 70 I=1,LIM
               J = LLK+I-1
        70     U(J) = K
     C      ASSIGNMENT OF ORBITAL EXPONENTS TO ATOMS BY SLATERS RULES
               MU(2)=1.700
               MU(1)=1.200
               NC(1)=1
               NC(2)=1
               DO 80 I=3,10
               NC(I)=2
        80     MU(I)=.3250*(I-1)
               DO 90 I=11,18
```

```
N IV  RF04      PROGRAM CNINDO  00              SOURCE      1

    LABEL CONT  ...1..../....2..../....3..../....4..../....5..../....6..../....7
                NC(I)=3
     90         MU(I)=(.65D0*(I)-4.95D0)/3.D0
    C       ASSIGNMENT OF ANGULAR MOMENTUM QUANTUM NOS. TO ATOMIC ORBITALS
                LC(1)=0
                LC(2)=1
                LC(3)=1
                LC(4)=1
                LC(5)=2
                LC(6)=2
                LC(7)=2
                LC(8)=2
                LC(9)=2
                MC(1)=0
                MC(2)=1
                MC(3)=-1
                MC(4)=0
                MC(5)=0
                MC(6)=1
                MC(7)=-1
                MC(8)=2
                MC(9)=-2
    C       STEP THRU PAIRS OF ATOMS
                DO 320 K=1,NATOMS
                DO 320 L=K,NATOMS
                DO 100 I=1,3
                C1(I) = C(K,I)
     100        C2(I) = C(L,I)
    C       CALCULATE UNIT VECTOR ALONG INTERATOM AXIS,E
                CALL RELVEC(R,E,C1,C2)
                LLK = LLIM(K)
                LLL = LLIM(L)
                ULK = ULIM(K)
                ULL = ULIM(L)
                NORBK=ULK-LLK+1
                NORBL=ULL-LLL+1
                ANK=AN(K)
                ANL=AN(L)
    C       LOOP THRU PAIRS OF BASIS FUNCTIONS, ONE ON EACH ATOM
                DO 200 I=1,NORBK
                DO 200 J=1,NORBL
                IF(K.EQ.L) GO TO 150
     110        IF(MC(I).NE.MC(J)) GO TO 150
     120        IF(MC(I).LT.0) GO TO 140
                NEWV= ((MU(ANK)*R)**(2*NC(ANK)+1)*(MU(ANL)*R)**(2*NC(ANL)+1)/
           1    (FACT(2*NC(ANK))*FACT(2*NC(ANL))))
                NEWS = SS(NC(ANK),LC(I),MC(I),NC(ANL),LC(J),MU(ANK)*R,MU(ANL)*R)
                IF (NEWV .EQ. 0) GO TO 135
     130        PAIRS (I,J)=DSQRT (NEWV)*(-1.D0)**(LC(J)+MC(J))*NEWS
                GO TO 190
     135        PAIRS (I,J)=0.D0
                GO TO 190
     140        PAIRS(I,J)=PAIRS(I-1,J-1)
                GO TO 190
```

```
IV  RF04     PROGRAM CNINDO  DO                SOURCE      1                    D

   LABEL CONT  ...1....+....2....+....3....+....4....+....5....+....6....+....7..
     150       PAIRS(I,J)=0.0D0
               GO TO 190
     160       IF (I.EQ.J) GO TO 170
     180       PAIRS(I,J)=0.0D0
               GO TO 190
     170       PAIRS(I,J)=1.0D0
     190       CONTINUE
     200       CONTINUE
               LCJL=LC(NORBK)
               LCJLL=LC(NORBL)
               MAXL=MAXO(LCULK,LCJLL)
               IF(R.GT.0.00000100) GO TO 220
     210       GO TO 250
   C      ROTATE INTEGRALS FROM DIATOMIC BASIS TO MOLECULAR BASIS
     220       CALL HARMTR(T,MAXL,E)
               DO 230 I=1,NORBK
               DO 230 J=1,NORBL
               TEMP(I,J) = 0.D0
               DO 230 KK=1,NORBL
               TEMP(I,J) = TEMP(I,J)+T(J,KK)*PAIRS(I,KK)
     230       CONTINUE
               DO 240 I=1,NORBK
               DO 240 J=1,NORBL
               PAIRS(I,J) = 0.D0
               DO 240 KK=1,NORBK
               PAIRS(I,J) = PAIRS(I,J)+T(I,KK)*TEMP(KK,J)
     240       CONTINUE
   C      FILL S MATRIX
     250       CONTINUE
               DO 260 I=1,NORBK
               LLKP=LLK+I-1
               DO 260 J=1,NORBL
               LLLP=LLL+J-1
     260       S(LLKP,LLLP)=PAIRS(I,J)
   C      COMPUTATION OF 1-CENTER COULOMB INTEGRALS OVER SLATER S FUNCTIONS
               N1=NC(ANK)
               N2=NC(ANL)
               K1=MJ(ANK)
               K2=MJ(ANL)
               IF(K.NE.L) GO TO 290
     270       TERM1 = FACT(2*N1-1)/((2.D0*K2)**(2*N1))
               TERM2 = 0.D0
               LIM = 2*N1
               DO 280 J=1,LIM
               NUM =J*(2.D0*K1)**(2*N1-J)*FACT(4*N1-J-1)
               DEN = FACT(2*N1-J)*2.D0*N1*(2.D0*(K1+K2))**(4*N1-J)
               TERM2 = TERM2 + NUM/DEN
     280       CONTINUE
               GO TO 310
   C      COMPUTATION OF 2-CENTER COULOMB INTEGRALS OVER SLATER S FUNCTIONS
     290       TERM1=(R/2.D0)**(2*N2)*SS(0,0,0,2*N2-1,0.D0,2.D0*K2*R)
               TERM2 = 0.D0
               LIM = 2*N1
```

```
IV  RF04     PROGRAM CNINDO  DO                SOURCE     1                    DAT

  LABEL CONT  ...1..../....2..../....3..../....4..../....5..../....6..../....7...
              WRITE(3,303) K1, K2, TERM1
  303         FORMAT (/, 3HK1=, E10.3, 3X, 3HK2=, E10.3, 3X, 6HTERM1=, E10.3)
              WRITE(3,301)
    301       FORMAT(84H  M  J       N1 N2      FACT          R            ATERM       BTE
         1    RM       STERM       TERM2     )
              DO 300 J=1,LIM
              ATERM=J*(2.DO*K1)**(2*N1-J)*(R/2.DO)**(2*N1-J+2*N2)
              BTERM=FACT(2*N1-J)*2.DO*N1
              STERM=SS(2*N1-J,0,0,2*N2-1,0,2.DO*K1*R,2.DO*K2*R)
              TERM2=TERM2 + (ATERM/BTERM) * STERM
              FECT=FACT(2*N1 - J)
              WRITE(3,302)LIM,J,      N1,N2,FECT,R,ATERM,BTERM,STERM,TERM2
    302       FORMAT(2I3, 6X, 2I3, 6(1X, E10.3))
      300     CONTINUE
      310     GAMMA(K,L) = ((2.DO*K2)**(2*N2+1)/FACT(2*N2))*(TERM1-TERM2)
      320     CONTINUE
    C      SYMMETRIZATION OF OVERLAP AND COULOMB INTEGRAL MATRICES
              DO 330 I=1,N
              DO 330 J=I,N
      330     S(J,I) = S(I,J)
              DO 340 I=1,NATOMS
              DO 340 J=I,NATOMS
      340     GAMMA(J,I) = GAMMA(I,J)
              WRITE(3,350)
      350     FORMAT(1H1,1X,23HOVERLAP INTEGRAL MATRIX)
              CALL MATOUT(S,1)
    C      TRANSFER GAMMA TO 80X80 MATRIX P FOR PRINTING
              DO 360 I=1,NATOMS
              DO 360 J=1,NATOMS
      360     P(I,J)=GAMMA(I,J)
              WRITE(3,370)
      370     FORMAT(1X,23HCOULOMB INTEGRAL MATRIX)
              CALL MATOUT(NATOMS,2)
              RETURN
              END

ALLOCATION

    01E08 XX          03C10 DUMMY       01E08 P

ALLOCATION

    00004 CHARGE      00008 MULTIP      0000C AN          00034 BI

ALLOCATION

    00028 U           000A4 ULIM        000CC LLIM        000F4 NELECS
    000FC OCCB

ALLOCATION
```

```
FULL FORTRAN IV  RF04      PROGRAM CNINDO  00                SOURCE      1

                LABEL CONT  ...1..../....2..../....3..../....4..../....5..../....6.
00000 XXX       004D8 GAMMA     007F8 T         00A80 PAIRS     00D08 TEMP
00F90 C1        00FA8 C2        00FC0 YYY

/AUXINT   / ALLOCATION

00000 A         00088 B

/OPTION   / ALLOCATION

00000 OPTION    00004 OPNCLO    00008 HUCKEL    0000C CNDO      00010 INDO
00014 CLOSED    00018 OPEN

EQUIVALENCE ALLOCATION


CALLED  SUBROUTINE

:CNVIR      :CHS        RELVEC      FACT        QDBLE       QRTOID      SS
DSQRT       MAXO        HARMTR      QFWRIT      QIOED       QFINIO      QSNGL
QIOEI       QIOER       MATOUT      QSTOP

SCALAR  ALLOCATION

00100 INTGRL    00104 I         00108 K         0010C LLK       00110 JLK
00114 LIM       00118 J         0011C L         00120 R         00124 LLL
00128 ULL       0012C NORBK     00130 NORBL     00134 ANK       00138 ANL
0013C NEWV      00144 NEWS      0014C LCULK     00150 LCULL     00154 MAXL
00158 KK        0015C LLKP      00160 LLLP      00164 N1        00168 N2
0016C K1        00174 K2        00178 TERM1     00184 TERM2     0018C NUM
00194 DEN       0019C ATERM     001A0 BTERM     001A4 STERM     001A8 SFCT

ARRAY  ALLOCATION

001AC MU        0023C NC        00284 LC        002A8 MC        002CC E
```

```
IV  RE04      PROGRAM CNINDO  DO                SOURCE      1                    DATE

  LABEL CONT  ...1..../....2..../....3..../....4..../....5..../....6..../....7...

              DOUBLE PRECISION FUNCTION SS(NN1,LL1,MM,NN2,LL2,ALPHA,BETA)
              DOUBLE PRECISION S, XX, A, B, P, PT, X, PPP1, PPP2, PPP3, PPP
    C     PROCEDURE FOR CALCULATING REDUCED OVERLAP INTEGRALS
              COMMON/ARRAYS/S(31,31),XX(961),NZZ(87),NZZV(87),NZY(224),NZYV(224)
              COMMON/ADJXINT/A(17),B(17)
              INTEGER ULIM
              N1=NN1
              L1=LL1
              M=MM
              N2=NN2
              L2=LL2
              P =(ALPHA + BETA)/2.D0
              PT=(ALPHA - BETA)/2.D0
              X = 0.D0
              M=IABS(M)
    C     REVERSE QUANTUM NUMBERS IF NECESSARY
              IF((L2.LT.L1).OR.((L2.EQ.L1).AND.(N2.LT.N1))) GO TO 20
       10     GO TO 30
       20     K = N1
              N1= N2
              N2= K
              K= L1
              L1= L2
              L2= K
              PT=-PT
       30     CONTINUE
              K = MOD((N1+N2-L1-L2),2)
    C     FIND A AND B INTEGRALS
              CALL AINTGS(P,N1+N2)
              CALL BINTGS(PT,N1+N2)
              IF((L1.GT.0).OR.(L2.GT.0)) GO TO 60
    C     BEGIN SECTION USED FOR OVERLAP INTEGRALS INVOLVING S FUNCTIONS
    C     FIND Z TABLE NUMBER L
       40     L = (90+17*N1+N1**2+2*N2)/2
              WRITE(3,41) N1, N2, L
       41     FORMAT(/, 3I3)
              ULIM = N1+N2
              LLIM = 0
              DO 50 I=LLIM,ULIM
              Z=0
              NNI1=N1+N2-I+1
              DO 48 J=1,87
              IF (I+1+17*(L-1)-NZZ(J)) 48,42,48
       42     Z=NZZV(J)
              WRITE(3,43) I, Z
       43     FORMAT(I3, 3X, E10.3)
              GO TO 49
       48     CONTINUE
       49     CONTINUE
              X=X+Z*A(I+1)*B(NNI1)/2.D0
              WRITE(3, 44) X, Z, J, A(I+1), B(NNI1)
       44     FORMAT(2(E10.3, 3X), I3, 2(3X, E10.3))
```

```
IV  KF04      PROGRAM CNINDD  00                SOURCE      1                    DATE

   LABEL CONT  ...1....+....2....+....3....+....4....+....5....+....6....+....7...
      50       CONTINUE
               SS=X
               GO TO 80
   C      BEGIN SECTION USED FOR OVERLAPS INVOLVING NON-S FUNCTIONS
   C      FIND Y TABLE NUMBER L
      60       L=(5-M)*(24-10*M+M**2)*(83-30*M+3*M**2)/120+
            1     (30-9*L1+L1**2-2*N1)*(28-9*L1+L1**2-2*N1)/8+
            2     (30-9*L2+L2**2-2*N2)/2
               LLIM = 0
               DO 70 I=LLIM,8
               ULIM=4 - MOD(K+I,2)
               DO 70 J=LLIM,ULIM
               Y=0
               IIII=2*J+MOD(K+I,2)+1
               DO 68 I1=1,224
               IF ((I+1+9*J+9*5*(L-1))-NZY(I1)) 68,62,68
      62       Y=NZYV(I1)
               GO TO 69
      68       CONTINUE
      69       CONTINUE
               X=X+Y*A(I+1)*B(IIII)
      70       CONTINUE
               PPP1=(2*L1+1)*FACT(L1-M)
               PPP2=(2*L2+1)*FACT(L2-M)
               PPP3=(4.D0*FACT(L1+M)*FACT(L2+M))
               PPP=PPP1*PPP2/PPP3
               IF (PPP .GT. 0) GO TO 72
               WRITE (3, 101) PPP
               PPP=0
      72       SS=X*(FACT(M+1)/8.D0)**2*DSQRT(PPP)
      80       CONTINUE
     101       FORMAT (1X,4HPPP=, D15.6)
               RETURN
               END

ALLOCATION

     01E08 XX          03C10 NZZ          03D6C NZZV         03EC8 NZY

ALLOCATION

     00088 B

OUTINE

DBLE       IABS       MOD        AINTGS     BINTGS     :CHS
IDEI       QFIXID     :CNVIR     QIDER      QIOED      FACT
STOP

CATION
```

```
IV  RF04     PROGRAM CNINDO  00              SOURCE      1                    DAT

  LABEL CONT  ...1..../....2..../....3..../....4..../....5..../....6..../....7...

              SUBROUTINE HARMTR(T,MAXL,E)
              DOUBLE PRECISION T, E, COST,SINT,COSP,SINP, COS2T,SIN2T,COS2P,SIN2
         1    P,SQRT3
              DIMENSION T(9,9),E(3)
              COST = E(3)
              IF((1.D0-COST**2).GT.0.0000000001) GO TO 20
     10       SINT = 0.D0
              GO TO 30
     20       SINT=DSQRT(1.D0-COST**2)
     30       CONTINUE
              IF(SINT.GT.0.000001D0)  GO TO 50
     40       COSP = 1.D0
              SINP = 0.D0
              GO TO 70
     50       COSP = E(1)/SINT
     60       SINP = E(2)/SINT
     70       CONTINUE
              DO 80 I=1,9
              DO 80 J=1,9
     80       T(I,J) = 0.D0
              T(1,1) =1.D0
              IF (MAXL.GT.1) GO TO 100
     90       IF (MAXL.GT.0) GO TO 110
              GO TO 120
    100       COS2T = COST**2-SINT**2
              SIN2T = 2.D0*SINT*COST
              COS2P = COSP**2-SINP**2
              SIN2P = 2.D0*SINP*COSP
  C      TRANSFORMATION MATRIX ELEMENTS FOR D FUNCTIONS
              SQRT3=DSQRT(3.D0)
              T(5,5) = (3.D0*COST**2-1.D0)/2.D0
              T(5,6) = -SQRT3    *SIN2T/2.D0
              T(5,8) = SQRT3    *SINT**2/2.D0
              T(6,5) = SQRT3    *SIN2T*COSP/2.D0
              T(6,6) = COS2T*COSP
              T(6,7) = -COST*SINP
              T(6,8) =-T(6,5)/SQRT3
              T(6,9) = SINT*SINP
              T(7,5) = SQRT3    *SIN2T*SINP/2.D0
              T(7,6) = COS2T*SINP
              T(7,7) = COST*COSP
              T(7,8) = -T(7,5)/SQRT3
              T(7,9) = -SINT*COSP
              T(8,5) = SQRT3    *SINT**2*COS2P/2.D0
              T(8,6) = SIN2T*COS2P/2.D0
              T(8,7) = -SINT*SIN2P
               T(8,8) = (1.D0+COST**2)*COS2P/2.D0
              T(8,9) = -COST*SIN2P
              T(9,5) = SQRT3    *SINT**2*SIN2P/2.D0
              T(9,6) = SIN2T*SIN2P/2.D0
              T(9,7) = SINT*COS2P
              T(9,8) = (1.D0+COST**2)*SIN2P/2.D0
```

```
                LABEL CONT  ...1..../....2..../....3..../....4..../....5..../
                            T(3,9) = COST*COS2P
                110         CONTINUE
                C       TRANSFORMATION MATRIX ELEMENTS FOR P FUNCTIONS
                            T(2,2) = COST*COSP
                            T(2,3) = -SINP
                            T(2,4) = SINT*COSP
                            T(3,2) = COST*SINP
                            T(3,3) = COSP
                            T(3,4) = SINT*SINP
                            T(4,2) = -SINT
                            T(4,4) = COST
                120         CONTINUE
                            RETURN
                            END
```

CALLED SUBROUTINE

QGMYSU QDBLE DSQRT QSTOP

SCALAR ALLOCATION

```
000A4 HARMTR    000A8 COST      000BC SINT      000B8 COSP      000C0 SIN
000C8 I         000CC J         000D0 COS2T     000D8 SIN2T     000E0 COS
000E8 SIN2P     000F0 SQRT3
```

ARRAY ALLOCATION

```
FULL FORTRAN IV  RF04     PROGRAM CNINDD  00                SOURCE      1

                LABEL CONT  ...1..../. ..2..../....3..../....4..../....5..../

                            SUBROUTINE RELVEC(R,E,C1,C2)
                            DOUBLE PRECISION E, C1, C2,X,R
                            DIMENSION E(3),C1(3),C2(3)
                            X = D.DD
                            DO 10 I=1,3
                            E(I) = C2(I)-C1(I)
                            X=X+E(I)*E(I)
                     10     CONTINUE
                            R=DSQRT(X)
                            DO 40 I=1,3
                            IF (R.GT..DDDDD1DD) GO TO 30
                     20     GO TO 40
                     30     E(I) =E(I)/R
                     40     CONTINUE
                            RETURN
                            END

   CALLED  SUBROUTINE

   QGMYSU     DSQRT      QSTOP

   SCALAR  ALLOCATION

   00050 RELVEC     00054 X          00058 I

   ARRAY  ALLOCATION
```

```
                LABEL CONT  ...1..../....2..../....3..../....4..../....5..../
                            DOUBLE PRECISION FUNCTION FACT (N)
                            DOUBLE PRECISION PRODT
                            PRODT = 1.DD
                      20    DO 30 I=1,N
                      30    PRODT=PRODT*I
                      40    FACT=PRODT
                            RETURN
                            END
```

CALLED SUBROUTINE

QGMYSU :CNVIR QSTOP

SCALAR ALLOCATION

00020 FACT 00028 PRODT 00030 I

```
IV  RF04     PROGRAM CNINDO  00              SOURCE      1                    DAT

   LABEL CONT  ...1..../....2..../....3..../....4..../....5..../....6..../....7...

               SUBROUTINE BINTGS(X,K)
C      FILLS ARRAY OF B-INTEGRALS. NOTE THAT B(I) IS B(I-1) IN THE
C      USUAL NOTATION
C      FOR X.GT.3                          EXPONENTIAL FORMULA IS USED
C      FOR 2.LT.X.LE.3 AND K.GE.10         EXPONENTIAL FORMULA IS USED
C      FOR 2.LT.X.LE.3 AND K.GT.10         15 TERM SERIES IS USED
C      FOR 1.LT.X .E.2 AND K.GE.7          EXPONENTIAL FORMULA IS USED
C      FOR 1.LT.X.LE.2 AND K.GT.7          12 TERM SERIES IS USED
C      FOR .5.LT.X.LE.1 AND K.GE.5         EXPONENTIAL FORMULA IS USED
C      FOR .5.LT.X.LE.1 AND K.GT.5          7 TERM SERIES IS USED
C      FOR X.LE..5                          6 TERM SERIES IS USED
C      ************************************************************
               DOUBLE PRECISION A, B, ABSX, EXPX, EXPMX, Y, X
               COMMON/AUXINT/A(17),B(17)
               DO 5 I=1,17
               B(I)=0.DO
     5         CONTINUE
               IO=0
               ABSX=DABS(X)
               IF(ABSX.GT.3.DO) GO TO 120
    10         IF(ABSX.GT.2.DO) GO TO 20
    40         IF(ABSX.GT.1.DO) GO TO 50
    70         IF(ABSX.GT..5DO) GO TO 80
   100         IF(ABSX.GT..00000100) GO TO 110
               GO TO 170
   110         LAST=6
               GO TO 140
    80         IF(K.LE.5) GO TO 120
    90         LAST=7
               GO TO 140
    50         IF(K.LE.7) GO TO 120
    60         LAST=12
               GO TO 140
    20         IF(K.LE.10) GO TO 120
    30         LAST=15
               GO TO 140
   120         EXPX=DEXP(X)
               EXPMX=1.DO/EXPX
               B(1)=(EXPX-EXPMX)/X
               DO 130 I=1,K
   130         B(I+1)=(I*B(I)+(-1.DO)**I*EXPX-EXPMX)/X
               GO TO 190
   140         DO 160 I=IO,K
               Y=0.DO
               DO 150 M=IO,LAST
   150         Y=Y+(-X)**M*(1.DO-(-1.DO)**(M+I+1))/(FACT(M)*(M+I+1))
   160         B(I+1)=Y
               GO TO 190
   170         DO 180 I=IO,K
   180         B(I+1)=(1.DO-(-1.DO)**(I+1))/(I+1)
   190         CONTINUE
               RETURN
```

```
FULL FORTRAN IV  RF04      PROGRAM CNINDO  00                SOURCE      1

                    LABEL CONT  ...1..../....2..../....3..../....4..../....5..../
                                END

/AUXINT    / ALLOCATION

00000 A           00088 B

CALLED  SUBROUTINE

QGMYSU      DABS        DEXP        QRTOID      :CNVIR      FACT        QDBLE
QSTOP

SCALAR  ALLOCATION

000D4 BINTGS      000D8 I           000DC IO          000E0 ABSX        000E8 BAS
000EC EXPX        000F4 EXPMX       000FC Y           00104 M

ARRAY  ALLOCATION
```

```
RTRAN IV  RF04      PROGRAM CNINDO  00              SOURCE      1

      LABEL CONT  ...1..../....2..../....3..../....4..../....5..../....6..../...

                  SUBROUTINE AINTGS(X,K)
                  DOUBLE PRECISION A, B, X
                  COMMON/AUXINT/A(17),B(17)
                  DO 5 I=1,17
                  A(I)=0.DO
            5     CONTINUE
                  A(1) =DEXP(-X)/X
                  DO 10  I=1,K
           10     A(I+1) =(A(I)*I+DEXP(-X))/X
                  RETURN
                  END

  / ALLOCATION

        00088 B

 SUBROUTINE

    DEXP       :CNVIR     QSTOP

 ALLOCATION

INTGS     0004C I

ALLOCATION
```

```
RTRAN IV  RF04     PROGRAM CNINDD  00              SOURCE      1

      LABEL CONT  ...1..../....2..../....3..../....4..../....5..../....6..../...

                  SUBROUTINE MATOUT(N,MATOP)
                  DOUBLE PRECISION A
                  COMMON/ARRAYS/A(31,31,2),DUMMY(622)
                  DO 80 M=1,N,11
                  K=M+10
                  IF (K.LE.N) GO TO 30
            20    K=N
            30    CONTINUE
                  WRITE(3,40) (J,J=M,K)
            40    FORMAT(//,7X,11(4X,I2,3X),//)
                  DO 60 I=1,N
                  WRITE(3,50) I,(A(I,J,MATOP),J=M,K)
            50    FORMAT(1X,I2,4X,50(F9.4))
            60    CONTINUE
                  WRITE(3,70)
            70    FORMAT(//)
            80    CONTINUE
                  RETURN
                  END

 / ALLOCATION

        03C10 DUMMY

 SUBROUTINE

   QFWRIT     QIOEI      QFINIO     QIOED      QSTOP

 ALLOCATION

ATOUT     0004C M          00050 K          00054 J          00058 I

ALLOCATION
```

```
LABEL CONT ...1..../....2..../....3..../....4..../....5..../....6..../....7...
           SUBROUTINE HUCKCL
           DOUBLE PRECISION A, B, C, XXX, G, Q, YYY, ENERGY, XXY, RHO, TEMP
C     EXTENDED HUCKEL THEORY FOR CLOSED SHELLS
C     OVERLAPS ARE IN MATRIX A, COULOMB INTEGRALS (GAMMA) ARE IN MATRIX
           COMMON/ARRAYS/A(31,31),B(31,31),DUMMY(622)
           COMMON/INFO/NATOMS,CHARGE,MULTIP,AN(10),C(10,3),N
           COMMON/INFO1/CZ(10),U(31),ULIM(10),LLIM(10),NELECS,OCCA,OCCB
           COMMON/GAB/XXX(155),G(10,10),Q(31),YYY(31),ENERGY,XXY(312)
           COMMON/OPTION/OPTION,OPNCLO,HUCKEL,CNDO,INDO,CLOSED,OPEN
           DIMENSION ENEG(18,3),BETA0(18)
           DIMENSION G1(18),F2(18)
           INTEGER CHARGE,OCCA,OCCB,UL,AN,CZ,U,ULIM,ANI
           INTEGER OPTION,OPNCLO,HUCKEL,CNDO,INDO,CLOSED,OPEN
           G1(3)=.092012
           G1(4)=.1407
           G1(5)=.199265
           G1(6)=.267708
           G1(7)=.346029
           G1(8)=.43423
           G1(9)=.532305
           F2(3)=.049865
           F2(4)=.089125
           F2(5)=.13041
           F2(6)=.17372
           F2(7)=.219055
           F2(8)=.266415
           F2(9)=.31580
           ENEG(1,1)=7.1761
           ENEG(3,1)=3.1055
           ENEG(3,2)=1.258
           ENEG(4,1)=5.94557
           ENEG(4,2)=2.563
           ENEG(5,1)=9.59407
           ENEG(5,2)=4.001
           ENEG(6,1)=14.051
           ENEG(6,2)=5.572
           ENEG(7,1)=19.31637
           ENEG(7,2)=7.275
           ENEG(8,1)=25.39017
           ENEG(8,2)=9.111
           ENEG(9,1)=32.2724
           ENEG(9,2)=11.08
           ENEG(11,1)=2.804
           ENEG(11,2)=1.302
           ENEG(11,3)=0.150
           ENEG(12,1)=5.1254
           ENEG(12,2)=2.0516
           ENEG(12,3)=0.16195
           ENEG(13,1)=7.7705
           ENEG(13,2)=2.9951
           ENEG(13,3)=0.22425
           ENEG(14,1)=10.0327
```

```
FTRAN IV  RF04      PROGRAM CNINDO  DO                SOURCE       1

      LABEL CONT  ...1..../..../....2..../..../....3..../..../....4..../..../....5..../..../....6..../..../...
                  ENEG(14,2)=4.1325
                  ENEG(14,3)=0.337
                  ENEG(15,1)=14.0327
                  ENEG(15,2)=5.4638
                  ENEG(15,3)=0.500
                  ENEG(16,1)=17.648
                  ENEG(16,2)=6.989
                  ENEG(16,3)=0.71325
                  ENEG(17,1)=21.5906
                  ENEG(17,2)=8.7081
                  ENEG(17,3)=0.97695
                  BETAD(1)= -9.
                  BETAD(3)= -9.
                  BETAD(4)= -13.
                  BETAD(5)= -17.
                  BETAD(6)= -21.
                  BETAD(7)=-25.
                  BETAD(8)= -31.
                  BETAD(9)= -39.
                  BETAD(11)=-7.7203
                  BETAD(12)=-9.4471
                  BETAD(13)=-11.301
                  BETAD(14)=-13.065
                  BETAD(15)=-15.070
                  BETAD(16)=-18.150
                  BETAD(17)=-22.330
      C       FIND NELECS AND FILL H CORE(DIAGONAL) WITH (I+A)/2
                  NELECS=0
                  DO 60 I=1,NATOMS
                  NELECS=NELECS+CZ(I)
                  LL =LLIM(I)
                  UL =ULIM(I)
                  ANI=AN(I)
                  L=0
                  DO 50 J=LL,UL
                  L=L+1
                  IF (L.EQ.1) GO TO 10
            20    IF (L.LT.5) GO TO 40
            30    A(J,J)=-ENEG(ANI,3)/27.21D0
                  GO TO 50
            40    A(J,J)=-ENEG(ANI,2)/27.21D0
                  GO TO 50
            10    A(J,J)  =-ENEG(ANI,1)/27.21D0
            50    CONTINUE
            60    CONTINUE
                  NELECS=NELECS-CHARGE
                  OCCA=NELECS/2
      C       FORM HUCKEL HAMILTONIAN IN A (OFF DIAGONAL TWO CENTER TERMS)
                  DO 90 I=2,N
                  K=J(I)
                  L=AN(K)
                  UL=I-1
                  DO 90 J=1,JL
```

```
      LABEL CONT  ...1....+....2....+....3....+....4....+....5....+....6....+...
                  KK=J(J)
                  LL=AN(KK)
                  IF ((L.GT.9).OR.(LL.GT.9)) GO TO 70
        80        A(I,J)=A(I,J)*(BETAO(L)+BETAO(LL))/54.42D0
                  A(J,I)=A(I,J)
                  GO TO 90
        70        A(I,J)=0.75D0*A(I,J)*(BETAO(L)+BETAO(LL))/54.42D0
                  A(J,I)=A(I,J)
        90        CONTINUE
                  DO 100 I=1,N
       100        Q(I)=A(I,I)
                  RHO=1.D-6
                  CALL EIGN(N,RHO)
      C       EIGENVECTORS (IN B) ARE CONVERTED INTO DENSITY MATRIX (IN B)
                  DO 140 I=1,N
                  DO 120 J=I,N
                  XXX(J)=0.0D0
                  DO 110 K=1,OCCA
       110        XXX(J)= XXX(J)+2.D0*B(I,K)*B(J,K)
       120        CONTINUE
                  DO 130 J=I,N
       130        B(I,J)= XXX(J)
       140        CONTINUE
                  DO 150 I=1,N
                  DO 150 J=I,N
       150        B(J,I)=B(I,J)
      C       ADD V(AB) TO HCORE=CNDO
                  DO  170  I=1,N
                  J=J(I)
                  Q(I)=Q(I)  +0.5D0*B(J,J)
                  DO 160 K=1,NATOMS
       160        Q(I)=Q(I)-(CZ(K))*B(J,K)
       170        CONTINUE
      C       EXIT SEGMENT IF ONLY CNDO APPROXIMATIONS ARE DESIRED
                  IF (OPTION.EQ.CNDO) GO TO 290
      C       INDO MODIFICATION (CORRECTION TO U(I,I) )
       180        DO 280 I=1,NATOMS
                  K=AN(I)
                  J=LIM(I)
                  IF ((K.GT.1).AND.(K.LT.10)) GO TO 190
                  GO TO 280
       190        IF (K.LE.3) GO TO 210
       200        Q(J) =Q(J)+(CZ(I)-1.5D0)*G1(K)/6.D0
       210        IF(K.EQ.3) GO TO 220
       230        IF(K.EQ.4) GO TO 240
       250        TEMP = G1(K)/3.D0+(CZ(I)-2.5D0)*2.D0*F2(K)/25.D0
                  GO TO 260
       240        TEMP=G1(K)/4.D0
                  GO TO 260
       220        TEMP=G1(K)/12.D0
       260        CONTINUE
                  DO 270 L=1,3
                  KK=K+L
```

```
RTRAN IV  RF04      PROGRAM CNINDO  00              SOURCE      1

        LABEL CONT  ...1....+....2....+....3....+....4....+....5....+....6....+...
          270       Q(KKKK)=Q(KKKK)+TEMP
          280       CONTINUE
          290       CONTINUE
                    DO 310 I=1,N
                    DO 300 J=I,N
          300       A(J,I)=A(I,J)
          310       A(I,I)=Q(I)
                    WRITE(3,320)
          320       FORMAT(1X,18H CORE HAMILTONIAN /)
                    CALL SCFOUT(0,1)
                    RETURN
                    END

   / ALLOCATION

          01E08 B           03C10 DUMMY

   / ALLOCATION

ATOMS     00004 CHARGE      00008 MULTIP      0000C AN          00034 C

   / ALLOCATION

Z         00028 U           000A4 ULIM        000CC LLIM        000F4 NELECS
CCA       000FC OCCB

   / ALLOCATION

XX        004D8 G           007F8 Q           008F0 YYY         009E8 ENERGY
XY

   / ALLOCATION

PTION     00004 OPNCLD      00008 HUCKEL      0000C CNDO        00010 INDO
LOSED     00018 OPEN

 SUBROUTINE

    EIGN        ICNVIR      QFWRIT      QFINID      SCFOUT      QSTOP

 ALLOCATION

UCKCL     00228 I           0022C LL          00230 UL          00234 ANI
          0023C J           00240 K           00244 KK          00248 RHO
EMP       00258 KKK

ALLOCATION

NEG       00334 BETA0       0037C G1          003C4 F2
```

```
RTRAN IV  RF04     PROGRAM CNINDO  00              SOURCE      1

      LABEL CONT  ...1..../....2..../....3..../....4..../....5..../....6..../...
                  SUBROUTINE SCFCLD
                  DOUBLE PRECISION A, B, XXX, G, Q, YYY, ENERGY, XXY, C, RHO, PA
                 1  , OLDENG
      C     CNDO/INDO CLOSED SHELL SCF SEGMENT
      C     GAMMA MATRIX CONTAINED IN G, CORE HAMILTONIAN CONTAINED IN Q AND
      C     UPPER TRIANGLE OF A, AND INITIAL DENSITY MATRIX CONTAINED IN B
      C     OPTIONS   CNDO OR INDO
                  COMMON/ARRAYS/A(31,31),B(31,31),DUMMY(622)
                  COMMON/GAB/XXX(155),G(10,10),Q(31),YYY(31),ENERGY,XXY(312)
                  COMMON/INFO/NATOMS,CHARGE,MULTIP,AN(10),C(10,3),N
                  COMMON/INFO1/CZ(10),U(31),ULIM(10),LLIM(10),NELECS,OCCA,OCCB
                  COMMON/OPTION/OPTION,OPNCLO,HUCKEL,CNDO,INDO,CLOSED,OPEN
                  INTEGER  OPTION,OPNCLO,HUCKEL,CNDO,INDO,CLOSED,OPEN
                  INTEGER CHARGE,OCCA,OCCB,UL,ULIM,U,AN,CZ,Z
                  DIMENSION G1(18),F2(18)
                  G1(3)=.092012
                  G1(4)=.1407
                  G1(5)=.199265
                  G1(6)=.267708
                  G1(7)=.346029
                  G1(8)=.43423
                  G1(9)=.532305
                  F2(3)=.049865
                  F2(4)=.089125
                  F2(5)=.13041
                  F2(6)=.17372
                  F2(7)=.219055
                  F2(8)=.266415
                  F2(9)=.31580
                  Z=0
                  IT=10
                  RHO=1.D-6
         10       CONTINUE
                  Z = Z+1
                  ENERGY = 0.DO
      C     TRANSFER CORE HAMILTONIAN TO LOWER TRIANGLE OF A
                  DO 20 I=1,N
                  A(I,I)=Q(I)
                  DO 20 J=I,N
         20       A(J,I)=A(I,J)
                  DO 30 I=1,N
                  II=U(I)
                  A(I,I)=A(I,I)-B(I,I)*G(II,II)*0.5DO
                  DO 30 K=1,N
                  JJ=U(K)
         30       A(I,I)=A(I,I)+B(K,K)*G(II,JJ)
                  NM=N-1
                  DO 40 I=1,NM
                  II=U(I)
                  LL=I+1
                  DO 40 J=LL,N
                  JJ=U(J)
```

```
LABEL CONT  ...1..../....2..../....3..../....4..../....5..../....6..../....7..
   40       A(J,I)=A(J,I)=B(J,I)*G(II,JJ)*0.5D0
C      INDO MODIFICATION
            IF (OPTION.EQ.CNDO) GO TO 90
   50       DO 80 II=1,NATOMS
            K=AN(II)
            I=LIM(II)
            IF (K.EQ.1) GO TO 80
   60       PAA=B(I,I)+B(I+1,I+1)+B(I+2,I+2)+B(I+3,I+3)
            A(I,I)=A(I,I)=(PAA-B(I,I)) *G1(K)/6.D0
            DO 70 J=1,3
            LLL=I+J
            A(LLL,LLL)=A(LLL,LLL)=B(I,I)*G1(K)/6.D0-(PAA-B(I,I))*7.D0*
        1   F2(K)/50.D0+B(LLL,LLL)*11.D0*F2(K)/50.D0
   70       A(LLL,I)=A(LLL,I)+B(I,LLL)*G1(K)/2.D0
            I1=I+1
            I2=I+2
            I3=I+3
            A(I2,I1)=A(I2,I1)+B(I2,I1)*11.D0*F2(K)/50.D0
            A(I3,I1)=A(I3,I1)+B(I3,I1)*11.D0*F2(K)/50.D0
            A(I3,I2)=A(I3,I2)+B(I3,I2)*11.D0*F2(K)/50.D0
   80       CONTINUE
   90       CONTINUE
            DO 100 I=1,N
  100       ENERGY=ENERGY+0.5D0*B(I,I)*(A(I,I)+G(I))
            DO 105 I=1,NM
            LL=I+1
            DO 105 J=LL,N
  105       ENERGY=ENERGY+B(I,J)*(A(I,J)+A(J,I))
            WRITE(3,110) ENERGY
  110       FORMAT(//,10X,22H ELECTRONIC ENERGY    ,F12.6)
            IF(DABS(ENERGY-OLDENG).GE..00001D0) GO TO 150
  120       Z=11
  130       WRITE(3,140)
  140       FORMAT(5X,18H ENERGY SATISFIED /)
            GO TO 170
  150       CONTINUE
  160       OLDENG=ENERGY
  170       CONTINUE
            IF (Z.LE.IT) GO TO 210
C       SYMMETRIZE F FOR PRINTING (MATRIX A)
  180       DO 190 I=1,N
            DO 190 J=I,N
  190       A(I,J)=A(J,I)
            WRITE(3,200)
  200       FORMAT(1X,27H HARTREE-FOCK ENERGY MATRIX)
            CALL SCFOUT(0,1)
  210       CONTINUE
            CALL EIGN(N,RHO)
            IF (Z.LE.IT) GO TO 240
  220       WRITE(3,230)
  230       FORMAT(1X,28HEIGENVALUES AND EIGENVECTORS)
            CALL SCFOUT(1,2)
  240       CONTINUE
```

```
LABEL CONT  ...1..../....2..../....3..../....4..../....5..../....6..../....7..
C     EIGENVECTORS (IN B) ARE CONVERTED INTO DENSITY MATRIX (IN B)
                DO 280 I=1,N
                DO 260 J=I,N
                XXX(J)=0.0D0
                DO 250 K=1,OCCA
      250       XXX(J)= XXX(J)+B(I,K)*B(J,K)*2.0D0
      260       CONTINUE
                DO 270 J=I,N
      270       B(I,J)= XXX(J)
      280       CONTINUE
                DO 290 I=1,N
                DO 290 J=I,N
      290       B(J,I)=B(I,J)
                IF (Z.LE.IT) GO TO 10
      300       CONTINUE
                RETURN
                END
```

ALLOCATION

```
      01E08 B           03C10 DUMMY
```

ALLOCATION

```
      004D8 G           007F8 Q           008F0 YYY         009E8 ENERGY
```

ALLOCATION

```
      00004 CHARGE      00008 MULTIP      0000C AN          00034 C
```

ALLOCATION

```
      00028 U           000A4 ULIM        000CC LLIM        000F4 NELECS
      000FC OCCB
```

ALLOCATION

```
      00004 OPNCLO      00008 HUCKEL      0000C CNDO        00010 INDO
      00018 OPEN
```

OUTINE

```
FWRIT     QIOED     QFINIO    DABS      SCFOUT    EIGN
```

OCATION

```
      000FC Z           00100 IT          00104 RHO         0010C I
      00114 II          00118 K           0011C JJ          00120 NM
      00128 PAA         00130 LLL         00134 I1          00138 I2
```

```
IV  RF04     PROGRAM CNINDO  DD                SOURCE       1                    DA

   LABEL CONT  ...1..../....2..../....3..../....4..../....5..../....6..../....7..
               SUBROUTINE CPRINT
               DOUBLE PRECISION A, B, XXX, G, Q, YYY, ENERGY, XXY, C, DPM, DM,
             1 DMSP, DMPD, ATENG, RAD, TCHG, SLTR1, FACTOR, DP
     C       CNDO=INDO SCF CLOSED SHELL PRINTOUT SEGMENT
               COMMON/ARRAYS/A(31,31),B(31,31),DUMMY(622)
               COMMON/GAB/XXX(155),G(10,10),Q(31),YYY(31),ENERGY,XXY(312)
               COMMON/INFO/NATOMS,CHARGE,MULTIP,AN(10),C(10,3),N
               COMMON/INFO1/CZ(10),U(31),ULIM(10),LLIM(10),NELECS,OCCA,OCCB
               COMMON/PERTBL/EL(18)
               COMMON/OPTION/OPTION,OPNCLO,HUCKEL,CNDO,INDO,CLOSED,OPEN
               INTEGER  OPTION,OPNCLO,HUCKEL,CNDO,INDO,CLOSED,OPEN
               INTEGER CHARGE,AN,J,ULIM,EL,OCCA,OCCB,UL,CZ,ANI
               DIMENSION DPM(3),DM(3),DMSP(3),DMPD(3)
               DIMENSION ATENG(18)
               IF (OPTION.EQ.CNDO) GO TO 20
               ATENG(1)=-0.6387302462  D0
               ATENG(3)=-.2321972405   D0
               ATENG(4)=-1.1219520354  D0
               ATENG(5)=-2.8725750048  D0
               ATENG(6)=-5.9349548261  D0
               ATENG(7)=-10.6731741251 D0
               ATENG(8)=-17.2920850650 D0
               ATENG(9)=-26.2574377875 D0
               GO TO 30
        20     CONTINUE
               ATENG(1)=-0.6387302462  D0
               ATENG(3)=-.2321972405   D0
               ATENG(4)=-1.1454120355  D0
               ATENG(5)=-2.9774239048  D0
               ATENG(6)=-5.1649936261  D0
               ATENG(7)=-11.0768746252 D0
               ATENG(8)=-18.0819658651 D0
               ATENG(9)=-27.5491302880 D0
               ATENG(11)=-.1977009568  D0
               ATENG(12)=-.8671913833  D0
               ATENG(13)=-2.0364557744 D0
               ATENG(14)=-3.8979034686 D0
               ATENG(15)=-6.7966009163 D0
               ATENG(16)=-10.765817434lD0
               ATENG(17)=-16.046701794OD0
        30     CONTINUE
               K=NATOMS-1
               WRITE(3,40)
        40     FORMAT(1X,15H DENSITY MATRIX)
               CALL SCFOUT(0,2)
               DO 50 I=1,K
               L=I+1
               DO 50 J=L,NATOMS
               RAD=DSQRT((C(I,1)-C(J,1))**2+(C(I,2)-C(J,2))**2
             1           +(C(I,3)-C(J,3))**2)
        50     ENERGY =ENERGY +(CZ(I)*CZ(J))/RAD
               WRITE(3,60) ENERGY
```

```
LABEL CONT  ...1....+....2....+....3....+....4....+....5....+....6....+....7..
   60       FORMAT(//,10X,16H TOTAL ENERGY = F16.10)
            DO 70 I=1,NATOMS
            ANI=AN(I)
   70       ENERGY=ENERGY-ATENG(ANI)
            WRITE(3,80) ENERGY
   80       FORMAT(//,10X,16HBINDING ENERGY=,F16.10,5H A.U.,//)
            WRITE(3,85)
   85       FORMAT(//,10X,42H VALENCE ELECTRON POPULATION FOR EACH ATOM,//)
            DO 110 I=1,NATOMS
            TCHG = 0.D0
            LL=LLIM(I)
            UL=JLIM(I)
            DO 90 J=LL,UL
   90       TCHG = TCHG+B(J,J)
            ANI=AN(I)
            WRITE(3,100) I,ELE(ANI),TCHG
  100       FORMAT(I3,A4,8X,F7.4)
            XXX(I)=TCHG
  110       CONTINUE
            DO 120 I=1,3
            DM(I)=0.0D0
            DMSP(I)=0.0D0
  120       DMPD(I)=0.0D0
            DO 200 J=1,NATOMS
            IF (AN(J).GT.3) GO TO 180
  130       IF (AN(J).GT.11) GO TO 140
  160       SLTR1=(.65D0*AN(J)+4.95D0)/3.D0
            FACTOR=2.5416D0*7.D0/(DSQRT(5.D0)*SLTR1)
            INDEX=LLIM(J)
            DO 170 K=1,3
            MMMM=INDEX+K
  170       DMSP(K)=DMSP(K)+B(INDEX,MMMM)*10.27175D0/SLTR1
            DMPD(1)=DMPD(1)+FACTOR*(B(INDEX+2,INDEX+8)+B(INDEX+3,INDEX+5)
           1   +B(INDEX+1,INDEX+7)-1.D0/DSQRT(3.D0)*B(INDEX+1,INDEX+4))
            DMPD(2)=DMPD(2)+FACTOR*(B(INDEX+1,INDEX+8)+B(INDEX+3,INDEX+6)
           1   +B(INDEX+2,INDEX+7)-1.D0/DSQRT(3.D0)*B(INDEX+2,INDEX+4))
            DMPD(3)=DMPD(3)+FACTOR*(B(INDEX+1,INDEX+5)+B(INDEX+2,INDEX+6)
           1   +2.D0/DSQRT(3.D0)*B(INDEX+3,INDEX+4))
            GO TO 180
  140       INDEX=LLIM(J)
            DO 150 K=1,3
            MMMM=INDEX + K
  150       DMSP(K)=DMSP(K)+B(INDEX,MMMM)*7.33697D0/
           1  (.325D0*(AN(J)-1))
  180       DO 190 I=1,3
  190       DM(I)=DM(I)+(CZ(J)-XXX(J))*C(J,I)*2.541600
  200       CONTINUE
            DO 210 I=1,3
  210       DPM(I)=DM(I)+DMSP(I)+DMPD(I)
            WRITE (3,220)
  220       FORMAT(//,20X,16H  DIPOLE MOMENTS,/)
            WRITE(3,230)
  230       FORMAT(5X,11H COMPONENTS,3X,2H X,8X,2H Y,8X,2H Z)
```

```
IV  RF04     PROGRAM CNINDO  00              SOURCE      1                    DA

    LABEL CONT  ...1..../....2..../....3..../....4..../....5..../....6..../....7..
                WRITE(3,240)DM(1),DM(2),DM(3)
      240       FORMAT(5X,10H DENSITIES,3(1X,F9.5))
                WRITE(3,250)DMSP(1),DMSP(2),DMSP(3)
      250       FORMAT(5X,4H S-P,6X,3(1X,F9.5))
                WRITE(3,260)DMPD(1),DMPD(2),DMPD(3)
      260       FORMAT(5X,4H P-D,6X,3(1X,F9.5))
                WRITE(3,270)DPM(1),DPM(2),DPM(3)
      270       FORMAT(5X,6H TOTAL,4X,3(1X,F9.5),/)
                DP=DSQRT(DPM(1)**2+DPM(2)**2+DPM(3)**2)
                WRITE(3,280) DP
      280       FORMAT(3X,15H DIPOLE MOMENT=,F9.5,7H DEBYES,//)
                RETURN
                END

ALLOCATION

    01E08 B          03C10 DUMMY

ALLOCATION

    004D8 G          007F8 Q          008F0 YYY        009E8 ENERGY

ALLOCATION

    00004 CHARGE     00008 MULTIP     0000C AN         00034 C

ALLOCATION

    00028 U          00DA4 ULIM       000CC LLIM       000F4 NELECS
    000FC OCCB

ALLOCATION

ALLOCATION

    00004 OPNCLD     00008 HUCKEL     0000C CNDO-      00010 INDO
    00018 OPEN

OUTINE

FINIO     SCFOUT     DSQRT      :CNVIR     QIDED      QIDEI

CATION

    00204 K          00208 I          0020C L          00210 J
    0021C ANI        00220 TCHG       00228 LL         0022C UL
    00238 FACTOR     00240 INDEX      00244 MMMM       00248 DP
```

```
IV  RF04      PROGRAM CNINDD  00              SOURCE      1                    DA

   LABEL CONT  ...1..../....2..../....3..../....4..../....5..../....6..../....7..
               SUBROUTINE EIGN(NN,RHO)
               DOUBLE PRECISION RHO, A, VEC, GAMMA, BETA, BETASQ, EIG, W, XYZ,
             1 P, Q, RHOSQ, S, SGN, SQRTS, D, TEMP, WTAW, SUM, QJ, WJ, SHIFT,
             2 COSA, G, PPBS, PPBT, COSAP, SINA, SINA2, DIA, U, A2, R2, R1, R12,
             3 DIF, PP
     C       RHO= UPPER LIMIT FOR OFF-DIAGONAL ELEMENT
     C       NN= SIZE OF MATRIX
     C       A = F MATRIX (ONLY LOWER TRIANGLE IS USED + THIS IS DESTROYED)
     C       EIG = RETURNED EIGENVALUES IN ALGEBRAIC ASCENDING ORDER
     C       VEC = RETURNED EIGENVECTORS IN COLUMNS
               COMMON/ARRAYS/A(31,31),VEC(31,31),DUMMY(622)
               COMMON/GAB/GAMMA(31),BETA(31),BETASQ(31),EIG(31),W(31),XYZ(475)
     C       THE FOLLOWING DIMENSIONED VARIABLES ARE EQUIVALENCED
               DIMENSION P(31),Q(31)
               EQUIVALENCE (P(1),BETA(1)),(Q(1),BETA(1))
               DIMENSION IPOSV(31),IVPOS(31),IORD(31)
               EQUIVALENCE (IPOSV(1),GAMMA(1)),(IVPOS(1),BETA(1)),
             1 (IORD(1),BETASQ(1))
               RHOSQ=RHO*RHO
               N=NN
               IF (N .EQ. 0) GO TO 640
        10     N1=N-1
               N2=N-2
               GAMMA(1)=A(1,1)
               IF(N2) 200,190,40
        40     DO 180 NR=1,N2
               B=A(NR+1,NR)
               S=0.D0
               DO 50 I=NR,N2
        50     S=A(I+2,NR)*A(I+2,NR) +S
     C       PREPARE FOR POSSIBLE BYPASS OF TRANSFORMATION
               NRR=NR+1
               A(NRR,NR)=0.D0
               IF (S) 170,170,60
        60     S=S+B*B
               SGN=+1.D0
               IF (B) 70,80,80
        70     SGN=-1.D0
        80     SQRTS=DSQRT(S)
               D=SGN/(SQRTS+SQRTS)
               TEMP=DSQRT(.5D0+B*D)
               W(NR)=TEMP
               A(NRR,NR)=TEMP
               D=D/TEMP
               B=-SGN*SQRTS
     C       D IS FACTOR OF PROPORTIONALITY. NOW COMPUTE AND SAVE W VECTOR.
     C       EXTRA SINGLY SUBSCRIPTED W VECTOR USED FOR SPEED.
               DO 90 I=NR,N2
               TEMP=D*A(I+2,NR)
               II1=I+1
               W(II1)=TEMP
               II2=I+2
```

```
LABEL CONT  ...1..../....2..../....3..../....4..../....5..../....6..../....7...
   90       A(II2,NR)=TEMP
C      PREMULTIPLY VECTOR W BY MATRIX A TO OBTAIN P VECTOR.
C      SIMULTANEOUSLY ACCUMULATE DOT PRODUCT WP,(THE SCALAR K)
         WTAW=0.DO
         DO 140 I=NR,N1
         SUM=0.DO
         DO 100 J=NR,I
  100    SUM=SUM+A(I+1,J+1)*W(J)
         I1=I+1
         IF(N1-I1) 130,110,110
  110    DO 120 J=I1,N1
  120    SUM=SUM+A(J+1,I+1)*W(J)
  130    P(I)=SUM
  140    WTAW=WTAW+SUM*W(I)
C      P VECTOR AND SCALAR K  NOW STORED. NEXT COMPUTE Q VECTOR
         DO 150 I=NR,N1
  150    Q(I)=P(I)-WTAW*W(I)
C      NOW FORM PAP MATRIX, REQUIRED PART
         DO 160 J=NR,N1
         JJ1=J+1
         QJ=Q(J)
         WJ=W(J)
         DO 160 I=J,N1
         II1=I+1
  160    A(II1,JJ1)=A(I+1,J+1)-2.DO*(W(I)*QJ+WJ*Q(I))
  170    BETA(NR)=B
         BETASQ(NR)=B*B
  180    GAMMA(NRRR)=A(NR+1,NR+1)
  190    B=A(N,N-1)
         NN1= N-1
         BETA(NN1)=B
         BETASQ(NN1)=B*B
         GAMMA(N)=A(N,N)
  200    BETASQ(N)=0.DO
C      ADJOIN AN IDENTITY MATRIX TO BE POSTMULTIPLIED BY ROTATIONS.
         DO 220 I=1,N
         DO 210 J=1,N
  210    VEC(I,J)=0.DO
  220    VEC(I,I)=1.DO
         M=N
         SUM=0.DO
         NPAS=1
         GO TO 350
  230    SUM=SUM+SHIFT
         COSA=1.DO
         G=GAMMA(1)-SHIFT
         PP=G
         PPBS=PP*PP+BETASQ(1)
         PPBR=DSQRT(PPBS)
         DO 320 J=1,M
         COSAP=COSA
         IF (PPBS.GT.1.D-12) GO TO 250
  240    SINA=0.DO
```

```
LABEL CONT  ...1..../....2..../....3..../....4..../....5..../....6..../....7...
            SINA2=0.D0
            COSA=1.D0
            GO TO 290
  250       SINA=BETA(J)/PPBR
            SINA2=BETASQ(J)/PPBS
            COSA=PP/PPBR
C     POSTMULTIPLY IDENTITY BY P=TRANSPOSE MATRIX
            NT=J+NPAS
            IF(NT .LT. N) GO TO 270
  260       NT=N
  270       DO 280 I=1,NT
            TEMP=COSA*VEC(I,J)+SINA*VEC(I,J+1)
            JJ1=J+1
            VEC(I,JJ1)=-SINA*VEC(I,J)+COSA*VEC(I,J+1)
  280       VEC(I,J)=TEMP
  290       DIA=GAMMA(J+1)-SHIFT
            U=SINA2*(G+DIA)
            GAMMA(J)=G+U
            G=DIA-U
            PP=DIA*COSA-SINA*COSAP*BETA(J)
            IF(J .NE. M) GO TO 310
  300       BETA(J)=SINA*PP
            BETASQ(J)=SINA2*PP*PP
            GO TO 330
  310       PPBS=PP*PP+BETASQ(J+1)
            PPBR=DSQRT(PPBS)
            BETA(J)=SINA*PPBR
  320       BETASQ(J)=SINA2*PPBS
  330       MM1=M+1
            GAMMA(MM1)=G
C     TEST FOR CONVERGENCE OF LAST DIAGONAL ELEMENT
            NPAS=NPAS+1
            IF(BETASQ(M) .GT. RHOSQ) GO TO 370
  340       MM1=M+1
            EIG(MM1)=GAMMA(M+1)+SUM
  350       BETA(M)=0.D0
            BETASQ(M)=0.D0
            M=M-1
            IF(M .EQ. 0) GO TO 400
  360       IF(BETASQ(M) .LE. RHOSQ) GO TO 340
C     TAKE ROOT OF CORNER 2 BY 2 NEAREST TO LOWER DIAGONAL IN VALUE
C     AS ESTIMATE OF EIGENVALUE TO USE FOR SHIFT
  370       A2=GAMMA(M+1)
            R2=0.5D0*A2
            R1=0.5D0*GAMMA(M)
            R12=R1+R2
            DIF=R1-R2
            TEMP=DSQRT(DIF*DIF+BETASQ(M))
            R1=R12+TEMP
            R2=R12-TEMP
            DIF=DABS(A2-R1)-DABS(A2-R2)
            IF(DIF .LT. 0.D0) GO TO 390
  380       SHIFT=R2
```

```
   LABEL CONT  ...1....+....2....+....3....+....4....+....5....+....6....+....7..
               GO TO 230
     390       SHIFT=R1
               GO TO 230
     400       EIG(I)=GAMMA(I)+SUM
     C      INITIALIZE AUXILIARY TABLES REQUIRED FOR REARRANGING THE VECTORS
               DO 410 J=1,N
               IPOSV(J)=J
               IVPOS(J)=J
     410       IORD(J)=J
     C      USE A TRANSPOSITION SORT TO ORDER THE EIGENVALUES
               M=N
               GO TO 450
     420       DO 440 J=1,M
               IF (EIG(J) .LE. EIG(J+1)) GO TO 440
     430       TEMP=EIG(J)
               EIG(J)=EIG(J+1)
               JJ1=J+1
               EIG(JJ1)=TEMP
               ITEMP=IORD(J)
               IORD(J)=IORD(J+1)
               IORD(JJ1)=ITEMP
     440       CONTINUE
     450       M=M-1
               IF(M .NE. 0) GO TO 420
     460       IF(N1 .EQ. 0) GO TO 510
     470       DO 500 L=1,N1
               NV=IORD(L)
               NP=IPOSV(NV)
               IF(NP .EQ. L) GO TO 500
     480       LV=IVPOS(L)
               IVPOS(NP)=LV
               IPOSV(LV)=NP
               DO 490 I=1,N
               TEMP=VEC(I,L)
               VEC(I,L)=VEC(I,NP)
     490       VEC(I,NP)=TEMP
     500       CONTINUE
     510       CONTINUE
     C      BACK TRANSFORM THE VECTORS OF THE TRIPLE DIAGONAL MATRIX
               DO 570 NRR=1,N
               K=N1
     520       K=K-1
               IF(K .LE. 0) GO TO 560
     530       SUM=0.D0
               DO 540 I=K,N1
     540       SUM=SUM+VEC(I+1,NRR)*A(I+1,K)
               SUM=SUM+SUM
               DO 550 I=K,N1
               II1=I+1
     550       VEC(II1,NRR)=VEC(I+1,NRR)-SUM*A(I+1,K)
               GO TO 520
     560       CONTINUE
     570       CONTINUE
```

```
IV  RF04  .  PROGRAM CNINDO  00              SOURCE      1                    DAT

    LABEL CONT  ...1..../....2..../....3..../....4..../....5..../....6..../....7...
      640       RETURN
                END

ALLOCATION

     01E08 VEC          03C10 DUMMY

ALLOCATION

     000F8 BETA         001F0 BETASQ       002E8 EIG          003E0 W
     00000 IPOSV        000F8 P            000F8 Q            000F8 IVPOS

ALLOCATION


OUTINE

SNGL       QDBLE       DSQRT       DABS        QSTOP

CATION

     000BC RHOSQ        000C4 N            000C8 N1           000CC N2
     000D4 B            000D8 S            000E0 I            000E4 NRRR
     000F0 SQRTS        000F8 D            00100 TEMP         00108 III
     00110 WTAW         00118 SUM          00120 J            00124 I1
     0012C QJ           00134 WJ           0013C NN1          00140 M
     00148 SHIFT        00150 COSA         00158 G            00160 PP
     00170 PPBR         00178 COSAP        00180 SINA         00188 SINA2
     00194 DIA          0019C U            001A4 MM1          001A8 A2
     001B8 R1           001C0 R12          001C8 DIF          001D0 ITEMP
     001D8 NV           001DC NP           001E0 LV           001E4 NRR

ATION
```

```
LABEL CONT ...1..../....2..../....3..../....4..../....5..../....6..../....7..
                SUBROUTINE SCFOUT(OP,MOP)
                DOUBLE PRECISION A, XXX, C
C       THIS ROUTINE PRINTS THE ARRAY IN COMMON/ARRAYS/ WHICH IS DESIGNATE
C       MOP. IF OP = 1 THE EIGENVALUES CONTAINED IN COMMON/1/ ARE ALSO
C       PRINTED. IF OP= 0 THE EIGENVALUES ARE NOT PRINTED
                COMMON/ARRAYS/A(31,31,2),DUMMY(622)
                COMMON/GAB/XXX(630)
                COMMON/INFO/NATOMS,CHARGE,MULTIP,AN(10),C(10,3),N
                COMMON/INFO1/CZ(10),U(31),ULIM(10),LLIM(10),NELECS,OCCA,OCCB
                COMMON/ORB/ORB(9)
                COMMON/PERTBL/EL(18)
                INTEGER OP,AN,ANII,CZ,U,ORB,ULIM,EL,CHARGE,OCCA,OCCB
                INTEGER ORBO,NEL
                INTEGER DUMMY
                DIMENSION NEL(31),ORBO(31)
                EQUIVALENCE (NEL(1),DUMMY(1)),(ORBO(1),DUMMY(50))
                DO 120 M=1,N,11
                K=M+10
                IF (K.LE.N) GO TO 30
       20       K=N
       30       CONTINUE
                WRITE(3,100)
                IF (OP.EQ.1) GO TO 40
                GO TO 50
       40       CALL EIGOUT(M,K)
       50       CONTINUE
                IF (OP.EQ.1) GO TO 56
                DO 55 I=M,K
                II=J(I)
                ANII=AN(II)
                L=I-LLIM(II)+1
                NEL(I)=EL(ANII)
                ORBO(I)=ORB(L)
       55       CONTINUE
       56       CONTINUE
                WRITE(3,60) (I,I=M,K)
                IF (OP.EQ.1) GO TO 57
                WRITE(3,61) ((NEL(I),U(I)),I=M,K)
                WRITE(3,62) (ORBO(I),I=M,K)
       57       CONTINUE
                WRITE(3,63)
       60       FORMAT(13X,5019)
       61       FORMAT(17X,11(A4,I2,3X))
       62       FORMAT(17X,11(2X,A4,3X))
       63       FORMAT(//)
                DO 110 I=1,N
                II=J(I)
                ANII=AN(II)
                L=I-LLIM(II)+1
       70       WRITE(3,80) I,II,EL(ANII),ORB(L),(A(I,J,MOP),J=M,K)
       80       FORMAT(1X,I2,I3,A4,1X,A4,50(F9.4))
                IF (I.EQ.ULIM(II)) GO TO 90
```

```
LABEL CONT  ...1..../....2..../....3..../....4..../....5..../....6..../....7...
            GO TO 110
     90     WRITE(3,100)
    100     FORMAT(1X)
    110     CONTINUE
    120     CONTINUE
            IF (OP.NE.0.OR.MOP.NE.2) GO TO 300
            WRITE(3,130)
    130     FORMAT(///,10X,27H DIAGONAL OF DENSITY MATRIX,//)
            DO 200 I=1,N
            II=J(I)
            ANII=AN(II)
            L=I-LLIM(II)+1
            WRITE(3,150)I,II,E(ANII),ORB(L),A(I,I,MOP)
    150     FORMAT(1X,I2,I3,A4,1X,A4,F9.4)
    200     CONTINUE
            WRITE(3,220)
    220     FORMAT(//)
    300     CONTINUE
            WRITE(3,100)
            WRITE(3,100)
            RETURN
            END
```

ALLOCATION

03C10 DUMMY 03C10 NEL 03CD4 ORBO

ALLOCATION

ALLOCATION

00004 CHARGE 00008 MULTIP 0000C AN 00034 C

ALLOCATION

00028 J 000A4 ULIM 000CC LLIM 000F4 NELECS
000FC 0CC9

ALLOCATION

ALLOCATION

ALLOCATION

```
IV  RF04  _  PROGRAM CNINDO  00          SOURCE      1                    DAT

   LABEL CONT  ...1..../....2..../....3..../....4..../....5..../....6..../....7...

               SUBROUTINE EIGOUT(M,K)
               DOUBLE PRECISION XXX, EPSILN, YYY
   C      THIS ROUTINE IS CALLED IN SCFOUT TO PRINT THE EIGENVALUES M TO K
               COMMON/GAB/XXX(93),EPSILN(31),YYY(506)
               WRITE(3,10) (EPSILN(I),I=M,K)
      10       FORMAT(//,15H EIGENVALUES---,20(F9.4),//)
               RETURN
               END

ALLOCATION

     002E8 EPSILN      003ED YYY

OUTINE

FWRIT      QIOED       QFINIO      QSTOP

CATION

     00030 I

ATION
```

CLC≡CH
CNDO CLSD

4 ATOMS CHARGE = 0 MULTIPLICITY =1

6	0.0000000	0.0000000	0.0000000
6	1.2040000	0.0000000	0.0000000
1	2.2600000	0.0000000	0.0000000
17	-1.6320000	0.0000000	0.0000000

OVERLAP INTEGRAL MATRIX (continued on next page)

	1	2	3	4	5	6	7	8
1	1.0000	0.0000	0.0000	0.0000	0.5092	-0.4714	0.0000	0.0000
2	0.0000	1.0000	0.0000	0.0000	0.4714	-0.2917	0.0000	0.0000
3	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.3355	0.0000
4	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.3355
5	0.5092	0.4714	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000
6	-0.4714	-0.2917	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000
7	0.0000	0.0000	0.3355	0.0000	0.0000	0.0000	1.0000	0.0000
8	0.0000	0.0000	0.0000	0.3355	0.0000	0.0000	0.0000	1.0000
9	0.1003	0.1235	0.0000	0.0000	0.5380	0.4985	0.0000	0.0000
10	0.3219	-0.3520	0.0000	0.0000	0.0419	-0.0594	0.0000	0.0000
11	0.3634	-0.3491	0.0000	0.0000	0.0573	-0.0798	0.0000	0.0000
12	0.0000	0.0000	0.1787	0.0000	0.0000	0.0000	0.0143	0.0000
13	0.0000	0.0000	0.0000	0.1787	0.0000	0.0000	0.0000	0.0143
14	-0.1291	0.1051	0.0000	0.0000	-0.0258	0.0352	0.0000	0.0000
15	0.0000	0.0000	0.0000	0.2381	0.0000	0.0000	0.0000	0.0229
16	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
17	0.2236	-0.1821	0.0000	0.0000	0.0446	-0.0610	0.0000	0.0000
18	0.0000	0.0000	0.2381	0.0000	0.0000	0.0000	0.0229	0.0000

	12	13	14	15	16	17	18
1	0.0000	0.0000	-0.1291	0.0000	0.0000	0.2236	0.0000
2	0.0000	0.0000	0.1051	0.0000	0.0000	-0.1821	0.0000
3	0.1787	0.0000	0.0000	0.0000	0.0000	0.0000	0.2381
4	0.0000	0.1787	0.0000	0.2381	0.0000	0.0000	0.0000
5	0.0000	0.0000	-0.0258	0.0000	0.0000	0.0446	0.0000
6	0.0000	0.0000	0.0352	0.0000	0.0000	-0.0610	0.0000
7	0.0143	0.0000	0.0000	0.0000	0.0000	0.0000	0.0229
8	0.0000	0.0143	0.0000	0.0229	0.0000	0.0000	0.0000
9	0.0000	0.0000	-0.0029	0.0000	0.0000	0.0050	0.0000
10	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
11	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
12	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
13	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000
14	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000
15	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000
16	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000
17	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000
18	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000

COULOMB INTEGRAL MATRIX

	1	2	3	4
1	0.5903	0.3968	0.2336	0.3138
2	0.3968	0.5903	0.4430	0.1865
3	0.2336	0.4430	0.7500	0.1360
4	0.3138	0.1865	0.1360	0.5249

overlap integral matrix

9	10	11
0.1003	0.3219	0.3634
0.1235	-0.3520	-0.3491
0.0000	0.0000	0.0000
0.0000	0.0000	0.0000
0.5380	0.0419	0.0573
0.4985	-0.0594	-0.0798
0.0000	0.0000	0.0000
0.0000	0.0000	0.0000
1.0000	0.0048	0.0065
0.0048	1.0000	0.0000
0.0065	0.0000	1.0000
0.0000	0.0000	0.0000
0.0000	0.0000	0.0000
-0.0029	0.0000	0.0000
0.0000	0.0000	0.0000
0.0000	0.0000	0.0000
0.0050	0.0000	0.0000
0.0000	0.0000	0.0000

CORE HAMILTONIAN (continued on next page)

				1 C 1 S	2 C 1 PX	3 C 1 PY	4 C 1 PZ	5 C 2 S	6 C 2 PX
1	1	C	S	-6.6004	0.0000	0.0000	0.0000	-0.3930	0.3638
2	1	C	PX	0.0000	-6.2888	0.0000	0.0000	-0.3638	0.2251
3	1	C	PY	0.0000	0.0000	-6.2888	0.0000	0.0000	0.0000
4	1	C	PZ	0.0000	0.0000	0.0000	-6.2888	0.0000	0.0000
5	2	C	S	-0.3930	-0.3638	0.0000	0.0000	-5.9185	0.0000
6	2	C	PX	0.3638	0.2251	0.0000	0.0000	0.0000	-5.6069
7	2	C	PY	0.0000	0.0000	-0.2590	0.0000	0.0000	0.0000
8	2	C	PZ	0.0000	0.0000	0.0000	-0.2590	0.0000	0.0000
9	3	H	S	-0.0553	-0.0681	0.0000	0.0000	-0.2966	-0.2748
10	4	CL	S	-0.1922	0.2102	0.0000	0.0000	-0.0250	0.0355
11	4	CL	PX	-0.2170	0.2085	0.0000	0.0000	-0.0342	0.0476
12	4	CL	PY	0.0000	0.0000	-0.1067	0.0000	0.0000	0.0000
13	4	CL	PZ	0.0000	0.0000	0.0000	-0.1067	0.0000	0.0000
14	4	CL	DZ2	0.0771	-0.0628	0.0000	0.0000	0.0154	-0.0210
15	4	CL	DXZ	0.0000	0.0000	0.0000	-0.1422	0.0000	0.0000
16	4	CL	DYZ	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
17	4	CL	DX-Y	-0.1335	0.1088	0.0000	0.0000	-0.0266	0.0364
18	4	CL	DXY	0.0000	0.0000	-0.1422	0.0000	0.0000	0.0000

				12 CL 4 PY	13 CL 4 PZ	14 CL 4 DZ2	15 CL 4 DXZ	16 CL 4 DYZ	17 CL 4 DX-Y
1	1	C	S	0.0000	0.0000	0.0771	0.0000	0.0000	-0.1335
2	1	C	PX	0.0000	0.0000	-0.0628	0.0000	0.0000	0.1088
3	1	C	PY	-0.1067	0.0000	0.0000	0.0000	0.0000	0.0000
4	1	C	PZ	0.0000	-0.1067	0.0000	-0.1422	0.0000	0.0000
5	2	C	S	0.0000	0.0000	0.0154	0.0000	0.0000	-0.0266
6	2	C	PX	0.0000	0.0000	-0.0210	0.0000	0.0000	0.0364
7	2	C	PY	-0.0085	0.0000	0.0000	0.0000	0.0000	0.0000
8	2	C	PZ	0.0000	-0.0085	0.0000	-0.0136	0.0000	0.0000
9	3	H	S	0.0000	0.0000	0.0012	0.0000	0.0000	-0.0021
10	4	CL	S	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
11	4	CL	PX	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
12	4	CL	PY	-5.8692	0.0000	0.0000	0.0000	0.0000	0.0000
13	4	CL	PZ	0.0000	-5.8692	0.0000	0.0000	0.0000	0.0000
14	4	CL	DZ2	0.0000	0.0000	-5.5851	0.0000	0.0000	0.0000
15	4	CL	DXZ	0.0000	0.0000	0.0000	-5.5851	0.0000	0.0000
16	4	CL	DYZ	0.0000	0.0000	0.0000	0.0000	-5.5851	0.0000
17	4	CL	DX-Y	0.0000	0.0000	0.0000	0.0000	0.0000	-5.5851
18	4	CL	DXY	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Core Hamiltonian

7 C 2 PY	8 C 2 PZ	9 H 3 S	10 Cl 4 S	11 CL 4 PX
0.0000	0.0000	-0.0553	-0.1922	-0.2170
0.0000	0.0000	-0.0381	0.2102	0.2085
-0.2590	0.0000	0.0000	0.0000	0.0000
0.0000	-0.2590	0.0000	0.0000	0.0000
0.0000	0.0000	-0.2966	-0.0250	-0.0342
0.0000	0.0000	-0.2748	0.0355	0.0476
-5.6069	0.0000	0.0000	0.0000	0.0000
0.0000	-5.6069	0.0000	0.0000	0.0000
0.0000	0.0000	-4.2969	-0.0021	-0.0028
0.0000	0.0000	-0.0021	-6.3427	0.0000
0.0000	0.0000	-0.0028	0.0000	-5.8692
-0.0085	0.0000	0.0000	0.0000	0.0000
0.0000	-0.0085	0.0000	0.0000	0.0000
0.0000	0.0000	0.0012	0.0000	0.0000
0.0000	-0.0136	0.0000	0.0000	0.0000
0.0000	0.0000	0.0000	0.0000	0.0000
0.0000	0.0000	-0.0021	0.0000	0.0000
-0.0136	0.0000	0.0000	0.0000	0.0000

18 CL 4 DXY
0.0000
0.0000
-0.1422
0.0000
0.0000
0.0000
-0.0136
0.0000
0.0000
0.0000
0.0000
0.0000
0.0000
0.0000
0.0000
0.0000
-5.5851

ELECTRONIC ENERGY -55.952315

ELECTRONIC ENERGY -55.974674

ELECTRONIC ENERGY -55.977344

ELECTRONIC ENERGY -55.977701

ELECTRONIC ENERGY -55.977771

ELECTRONIC ENERGY -55.977784

ELECTRONIC ENERGY -55.977792

ENERGY SATISFIED

HARTREE-FOCK ENERGY MATRIX (continued on next page)

				1 C 1 S	2 C 1 PX	3 C 1 PY	4 C 1 PZ	5 C 2 S	6 C 2 PX	7 C 2 PY
1	1	C	S	-0.5382	0.0036	-0.0000	-0.0000	-0.4952	0.4722	0.0000
2	1	C	PX	0.0036	-0.1737	-0.0000	-0.0000	-0.4686	0.3019	-0.0000
3	1	C	PY	-0.0000	-0.0000	-0.1999	-0.0000	0.0000	-0.0000	-0.4491
4	1	C	PZ	-0.0000	-0.0000	-0.0000	-0.1999	0.0000	0.0000	-0.0000
5	2	C	S	-0.4952	-0.4686	0.0000	0.0000	-0.5562	-0.0127	-0.0000
6	2	C	PX	0.4722	0.3019	-0.0000	0.0000	-0.0127	-0.2044	-0.0000
7	2	C	PY	0.0000	-0.0000	-0.4491	-0.0000	-0.0000	-0.0000	-0.2074
8	2	C	PZ	-0.0000	-0.0000	-0.0000	-0.4491	0.0000	0.0000	-0.0000
9	3	H	S	-0.0498	-0.0633	0.0000	-0.0000	-0.4431	-0.4380	0.0000
10	4	CL	S	-0.2230	0.2528	0.0000	0.0000	-0.0256	0.0346	-0.0000
11	4	CL	PX	-0.3082	0.3144	-0.0000	0.0000	-0.0278	0.0407	0.0000
12	4	CL	PY	-0.0000	-0.0000	-0.1334	0.0000	0.0000	0.0000	0.0071
13	4	CL	PZ	-0.0000	0.0000	-0.0000	-0.1334	0.0000	-0.0000	-0.0000
14	4	CL	DZ2	0.0947	-0.0766	-0.0000	-0.0000	0.0157	-0.0225	-0.0000
15	4	CL	DXZ	-0.0000	-0.0000	-0.0000	-0.1781	0.0000	0.0000	-0.0000
16	4	CL	DYZ	0.0000	0.0000	-0.0000	0.0000	-0.0000	-0.0000	-0.0000
17	4	CL	DX-Y	-0.1640	0.1326	0.0000	0.0000	-0.0271	0.0389	0.0000
18	4	CL	DXY	-0.0000	-0.0000	-0.1781	0.0000	-0.0000	-0.0000	-0.0351

	12 CL 4	13 CL 4	14 CL 4	15 CL 4	16 CL 4	17 CL 4	18 CL 4

Hartree - Fock Energy Matrix (continued)

8 C 2 PZ	9 H 3 S	10 Cl 4 S	11 CL 4 PX
-0.0000	-0.0498	-0.2230	-0.3082
-0.0000	-0.0633	0.2528	0.3144
-0.0000	0.0000	0.0000	-0.0000
-0.4491	-0.0000	0.0000	0.0000
0.0000	-0.4431	-0.0256	-0.0278
0.0000	-0.4380	0.0346	0.0407
-0.0000	0.0000	-0.0000	0.0000
-0.2074	-0.0000	0.0000	0.0000
-0.0000	-0.2719	-0.0036	-0.0093
0.0000	-0.0036	-1.0052	0.0835
0.0000	-0.0093	0.0835	-0.3083
0.0000	-0.0000	-0.0000	-0.0000
0.0071	-0.0000	0.0000	0.0000
-0.0000	0.0014	-0.0030	0.0380
-0.0351	-0.0000	0.0000	0.0000
0.0000	-0.0000	-0.0000	-0.0000
0.0000	-0.0025	0.0052	-0.0658
0.0000	0.0000	-0.0000	-0.0000

Hartree Fock Energy Matrix

				PY	PZ	DZ2	DXZ	DYZ	DX-Y	DXY
1	1	C	S	-0.0000	-0.0000	0.0947	-0.0000	0.0000	-0.1640	-0.0000
2	1	C	PX	-0.0000	0.0000	-0.0766	-0.0000	0.0000	0.1326	-0.0000
3	1	C	PY	-0.1334	-0.0000	-0.0000	-0.0000	-0.0000	0.0000	-0.1781
4	1	C	PZ	0.0000	-0.1334	-0.0000	-0.1781	0.0000	0.0000	0.0000
5	2	C	S	0.0000	0.0000	0.0157	0.0000	-0.0000	-0.0271	-0.0000
6	2	C	PX	0.0000	-0.0000	-0.0225	0.0000	-0.0000	0.0389	-0.0000
7	2	C	PY	0.0071	-0.0000	-0.0000	-0.0000	-0.0000	0.0000	-0.0351
8	2	C	PZ	0.0000	0.0071	-0.0000	-0.0351	0.0000	0.0000	0.0000
9	3	H	S	-0.0000	-0.0000	0.0014	-0.0000	-0.0000	-0.0025	0.0000
10	4	CL	S	-0.0000	0.0000	-0.0030	0.0000	-0.0000	0.0052	-0.0000
11	4	CL	PX	-0.0000	0.0000	0.0380	0.0000	-0.0000	-0.0658	-0.0000
12	4	CL	PY	-0.5540	0.0000	-0.0000	0.0000	-0.0000	0.0000	0.0042
13	4	CL	PZ	0.0000	-0.5540	-0.0000	0.0042	0.0000	0.0000	-0.0000
14	4	CL	DZ2	-0.0000	-0.0000	0.2417	-0.0000	0.0000	0.0099	-0.0000
15	4	CL	DXZ	0.0000	0.0042	-0.0000	0.2332	0.0000	0.0000	0.0000
16	4	CL	DYZ	-0.0000	0.0000	0.0000	0.0000	0.2474	-0.0000	-0.0000
17	4	CL	DX-Y	0.0000	0.0000	0.0099	0.0000	-0.0000	0.2303	0.0000
18	4	CL	DXY	0.0042	-0.0000	-0.0000	0.0000	-0.0000	0.0000	0.2332

EIGENVALUES AND EIGENVECTORS (continued on next page)

EIGENVALUES---				-1.3873	-1.1510	-0.8964	-0.7264	-0.7264	-0.6746	-0.5175
				1	2	3	4	5	6	7
1	1	C	S	-0.6359	0.1272	-0.2866	-0.0000	-0.0000	0.1981	0.0000
2	1	C	PX	-0.1996	-0.4075	-0.0568	0.0000	0.0000	-0.4932	-0.0000
3	1	C	PY	0.0000	-0.0000	-0.0000	-0.4710	-0.4535	-0.0000	0.2353
4	1	C	PZ	-0.0000	0.0000	-0.0000	0.4535	-0.4710	0.0000	-0.1160
5	2	C	S	-0.6021	-0.3504	0.2597	-0.0000	0.0000	-0.0324	0.0000
6	2	C	PX	0.2455	-0.0839	0.5751	0.0000	-0.0000	0.2958	0.0000
7	2	C	PY	-0.0000	-0.0000	0.0000	-0.4097	-0.3945	-0.0000	0.3673
8	2	C	PZ	0.0000	0.0000	-0.0000	0.3945	-0.4097	0.0000	-0.1810
9	3	H	S	-0.1847	-0.2359	0.5580	0.0000	-0.0000	0.2471	-0.0000
10	4	CL	S	-0.2727	0.7844	0.3979	0.0000	0.0000	-0.3085	0.0000
11	4	CL	PX	-0.1344	0.1161	-0.2042	-0.0000	-0.0000	0.6665	0.0000
12	4	CL	PY	-0.0000	0.0000	-0.0000	-0.3451	-0.3322	-0.0000	-0.7799
13	4	CL	PZ	0.0000	-0.0000	-0.0000	0.3322	-0.3451	-0.0000	0.3844
14	4	CL	DZ2	0.0400	-0.0301	0.0355	-0.0000	-0.0000	-0.0845	0.0000
15	4	CL	DXZ	-0.0000	0.0000	-0.0000	0.0971	-0.1009	-0.0000	-0.0381
16	4	CL	DYZ	-0.0000	0.0000	0.0000	-0.0000	-0.0000	0.0000	0.0000
17	4	CL	DX-Y	-0.0693	0.0521	-0.0615	-0.0000	0.0000	0.1464	0.0000
18	4	CL	DXY	-0.0000	-0.0000	-0.0000	-0.1009	-0.0971	-0.0000	0.0774

Eigenvalues and Eigenvectors (continued)

-0.5175	0.0994	0.1560	0.1560	Eigenvalues
8	9	10	11	
-0.0000	-0.3856	0.0000	-0.0000	
0.0000	0.2988	-0.0000	0.0000	
0.1160	-0.0000	0.0970	0.4027	
0.2353	0.0000	0.4027	-0.0970	
-0.0000	0.1728	0.0000	-0.0000	
-0.0000	-0.0553	0.0000	-0.0000	
0.1810	-0.0000	-0.1360	-0.5646	
0.3673	-0.0000	-0.5646	0.1360	
0.0000	-0.1543	-0.0000	0.0000	
0.0000	0.1867	-0.0000	0.0000	
-0.0000	0.6414	-0.0000	0.0000	
-0.3844	-0.0000	-0.0186	-0.0773	
-0.7799	-0.0000	-0.0773	0.0186	
-0.0000	0.2545	-0.0000	0.0000	
0.0774	0.0000	0.6769	-0.1631	
0.0000	0.0000	0.0000	0.0000	
0.0000	-0.4408	0.0000	0.0000	
0.0381	0.0000	0.1631	0.6769	

Eigenvalues and Eigenvectors

EIGENVALUES===				0.2474	0.2474	0.2702	0.3509	0.3598	0.3598	0.5554
				12	13	14	15	16	17	18
1	1	C	S	0.0000	0.0000	0.2358	-0.3779	-0.0000	-0.0000	0.3331
2	1	C	PX	-0.0000	0.0000	0.1482	0.2928	0.0000	0.0000	0.5922
3	1	C	PY	0.0000	0.0000	-0.0000	0.0000	0.0796	-0.5707	-0.0000
4	1	C	PZ	-0.0000	0.0000	-0.0000	-0.0000	0.5707	0.0796	-0.0000
5	2	C	S	-0.0000	-0.0000	-0.5358	-0.1262	-0.0000	-0.0000	-0.3368
6	2	C	PX	-0.0000	-0.0000	-0.2672	-0.3214	-0.0000	-0.0000	0.5803
7	2	C	PY	0.0000	-0.0000	0.0000	-0.0000	-0.0572	0.4102	0.0000
8	2	C	PZ	0.0000	-0.0000	-0.0000	0.0000	-0.4102	-0.0572	0.0000
9	3	H	S	0.0000	0.0000	0.6171	0.3088	0.0000	0.0000	-0.1945
10	4	CL	S	-0.0000	0.0000	-0.0117	0.1275	0.0000	0.0000	0.0747
11	4	CL	PX	0.0000	-0.0000	0.0064	0.2332	0.0000	0.0000	0.1295
12	4	CL	PY	0.0000	-0.0000	-0.0000	-0.0000	-0.0125	0.0897	0.0000
13	4	CL	PZ	0.0000	-0.0000	0.0000	0.0000	-0.0897	-0.0125	0.0000
14	4	CL	DZ2	-0.8660	0.0008	0.2139	-0.3487	-0.0000	-0.0000	-0.0837
15	4	CL	DXZ	0.0000	-0.0000	0.0000	0.0000	-0.6920	-0.0965	0.0000
16	4	CL	DYZ	-0.0010	-1.0000	0.0000	-0.0000	0.0000	-0.0000	-0.0000
17	4	CL	DX-Y	-0.5000	0.0005	-0.3705	0.6039	0.0000	0.0000	0.1450
18	4	CL	DXY	0.0000	-0.0000	-0.0000	-0.0000	-0.0965	0.6920	0.0000

DENSITY MATRIX (continued on next page)

				1 C 1 S	2 C 1 PX	3 C 1 PY	4 C 1 PZ	5 C 2 S	6 C 2 PX	7 C 2 PY
1	1	C	S	1.0839	-0.0126	0.0000	0.0000	0.5150	-0.5461	-0.0000
2	1	C	PX	-0.0126	0.9047	0.0000	0.0000	0.5283	-0.3868	0.0000
3	1	C	PY	0.0000	0.0000	0.9926	-0.0000	-0.0000	0.0000	0.9585
4	1	C	PZ	0.0000	0.0000	-0.0000	0.9926	-0.0000	-0.0000	-0.0000
5	2	C	S	0.5150	0.5283	-0.0000	-0.0000	1.1075	0.0426	0.0000
6	2	C	PX	-0.5461	-0.3868	0.0000	-0.0000	0.0426	0.9712	0.0000
7	2	C	PY	-0.0000	0.0000	0.9585	-0.0000	0.0000	0.0000	0.9823
8	2	C	PZ	0.0000	0.0000	-0.0000	0.9585	-0.0000	-0.0000	0.0000
9	3	H	S	-0.0471	-0.0412	-0.0000	0.0000	0.6615	0.7369	0.0000
10	4	CL	S	0.1961	-0.2713	-0.0000	-0.0000	0.0054	0.0097	0.0000
11	4	CL	PX	0.5816	-0.6752	0.0000	-0.0000	-0.0687	0.0740	0.0000
12	4	CL	PY	-0.0000	0.0000	0.1702	-0.0000	-0.0000	-0.0000	-0.1673
13	4	CL	PZ	0.0000	-0.0000	0.0000	0.1702	0.0000	0.0000	-0.0000
14	4	CL	DZ2	-0.1124	0.0879	0.0000	0.0000	-0.0032	0.0155	0.0000
15	4	CL	DXZ	0.0000	0.0000	-0.0000	0.2284	-0.0000	-0.0000	-0.0000
16	4	CL	DYZ	0.0000	-0.0000	0.0000	-0.0000	-0.0000	0.0000	0.0000
17	4	CL	DX-Y	0.1946	-0.1522	0.0000	-0.0000	0.0055	-0.0269	0.0000
18	4	CL	DXY	0.0000	0.0000	0.2284	-0.0000	-0.0000	0.0000	0.2300

Density Matrix (continued)

8 C 2 PZ	9 H 3 S	10 Cl 4 S	11 CL 4 PX
0.0000	-0.0471	0.1961	0.5816
0.0000	-0.0412	-0.2713	-0.6752
-0.0000	-0.0000	-0.0000	0.0000
0.9585	0.0000	-0.0000	-0.0000
-0.0000	0.6615	0.0054	-0.0687
-0.0000	0.7369	0.0097	0.0740
0.0000	0.0000	0.0000	0.0000
0.9823	0.0000	-0.0000	-0.0090
0.0000	0.9244	0.0224	0.0963
-0.0000	0.0224	1.8863	-0.3182
-0.0000	0.0963	-0.3182	1.0349
0.0000	0.0000	0.0000	0.0000
-0.1673	-0.0000	-0.0000	-0.0000
-0.0000	-0.0027	0.0114	-0.1449
0.2300	0.0000	-0.0000	-0.0000
-0.0000	0.0000	-0.0000	0.0000
-0.0000	0.0047	-0.0197	0.2509
-0.0000	-0.0000	0.0000	0.0000

Density Matrix

				12 CL 4 PY	13 CL 4 PZ	14 CL 4 DZ2	15 CL 4 DXZ	16 CL 4 DYZ	17 CL 4 DX-Y	18 CL 4 DXY
1	1	C	S	-0.0000	0.0000	-0.1124	0.0000	0.0000	0.1946	0.0000
2	1	C	PX	0.0000	-0.0000	0.0879	0.0000	-0.0000	-0.1522	0.0000
3	1	C	PY	0.1702	0.0000	0.0000	-0.0000	0.0000	0.0000	0.2284
4	1	C	PZ	-0.0000	0.1702	0.0000	0.2284	-0.0000	-0.0000	-0.0000
5	2	C	S	-0.0000	0.0000	-0.0032	-0.0000	-0.0000	0.0055	-0.0000
6	2	C	PX	-0.0000	0.0000	0.0155	-0.0000	0.0000	-0.0269	0.0000
7	2	C	PY	-0.1673	-0.0000	0.0000	-0.0000	0.0000	0.0000	0.2300
8	2	C	PZ	0.0000	-0.1673	-0.0000	0.2300	-0.0000	-0.0000	-0.0000
9	3	H	S	0.0000	-0.0000	-0.0027	0.0000	0.0000	0.0047	-0.0000
10	4	CL	S	0.0000	-0.0000	0.0114	-0.0000	-0.0000	-0.0197	0.0000
11	4	CL	PX	0.0000	-0.0000	-0.1449	-0.0000	0.0000	0.2509	0.0000
12	4	CL	PY	1.9710	-0.0000	0.0000	-0.0000	-0.0000	-0.0000	-0.0158
13	4	CL	PZ	-0.0000	1.9710	0.0000	-0.0158	-0.0000	-0.0000	0.0000
14	4	CL	DZ2	0.0000	0.0000	0.0218	0.0000	-0.0000	-0.0378	0.0000
15	4	CL	DXZ	-0.0000	-0.0158	0.0000	0.0541	-0.0000	-0.0000	-0.0000
16	4	CL	DYZ	-0.0000	-0.0000	-0.0000	-0.0000	0.0000	0.0000	0.0000
17	4	CL	DX-Y	-0.0000	-0.0000	-0.0378	-0.0000	0.0000	0.0654	0.0000
18	4	CL	DXY	-0.0158	0.0000	0.0000	-0.0000	0.0000	0.0000	0.0541

DIAGONAL OF DENSITY MATRIX

1	1	C	S	1.0839
2	1	C	PX	0.9047
3	1	C	PY	0.9926
4	1	C	PZ	0.9926
5	2	C	S	1.1075
6	2	C	PX	0.9712
7	2	C	PY	0.9823
8	2	C	PZ	0.9823
9	3	H	S	0.9244
10	4	CL	S	1.8863
11	4	CL	PX	1.0349
12	4	CL	PY	1.9710
13	4	CL	PZ	1.9710
14	4	CL	DZ2	0.0218
15	4	CL	DXZ	0.0541
16	4	CL	DYZ	0.0000
17	4	CL	DX-Y	0.0654
18	4	CL	DXY	0.0541

TOTAL ENERGY = -30.7495908377

BINDING ENERGY= -1.7341715453 A.U.

VALENCE ELECTRON POPULATION FOR EACH ATOM

1	C	3.9738
2	C	4.0432
3	H	0.9244
4	CL	7.0586

DIPOLE MOMENTS

COMPONENTS	X	Y	Z
DENSITIES	1.03027	0.00000	0.00000
S.P	1.47228	-0.00000	0.00000
P.D	-1.18543	0.00000	-0.00000
TOTAL	1.31712	0.00000	-0.00000

DIPOLE MOMENT= 1.31712 DEBYES

APPENDIX B

Below is a table of standard bond lengths used in all calculations in this paper.

Except where noted, all values are from Pople and Beveridge, Approximate Molecular Orbital Theory, p. 111. The number after the symbol for the atom is the connectivity, i. e. the number of atoms to which that atom is bonded.

Single bonds

Bond	Length, A
C3-H1	1.08
C2-H1	1.06
C3-C3	1.46
C3-C2	1.45
C3-N3	1.40
C3-F1	1.33
C2-F1	1.30
C3-Cl1	2.00*
C2-Cl1	1.632*
C3-C3	1.51*↓

Double bonds

Bond	Length, A
C3-C3	1.32
C3-C2	1.31
C3-N2	1.32
C3-O1	1.22
C2-O1	1.16
N3-O1	1.24

Triple bonds

Bond	Length, A
C2-C2	1.20
C2-N1	1.16

Aromatic bonds

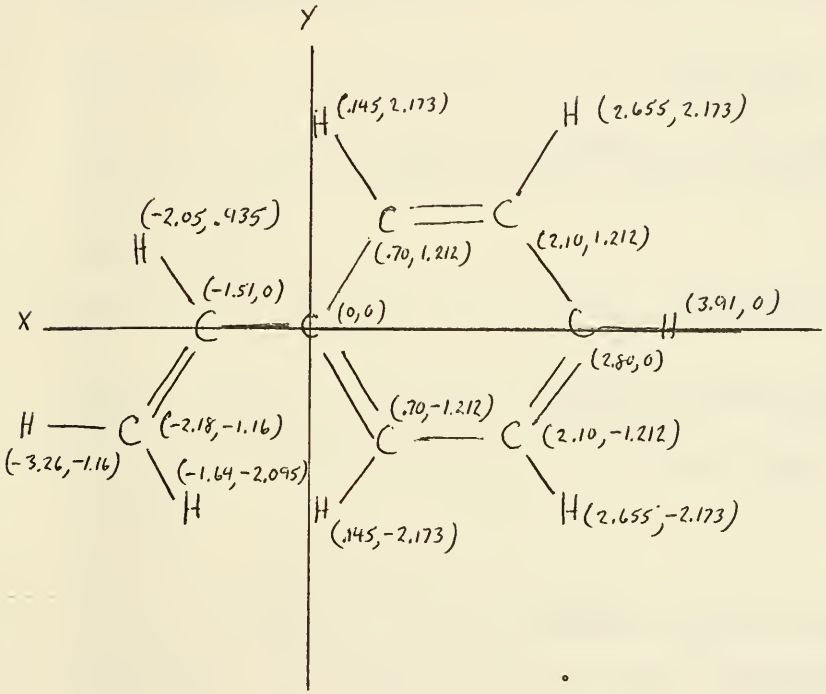
Bond	Length, A
C3-C3	1.40

* Sutton, L.E., Ed., Tables of Interatomic Distances and Configurations in Molecules and Ions, Special Publications Nos. 11 and 18, the Chemical Society, London (1958 and 1965).

↓ For C_1-C_7 bond in phenyl substituted molecules- taken from the value for toluene.

APPENDIX C

The diagram below shows how the phenyl-substituted molecules were arranged along a cartesian coordinate axis, and gives sample coordinates.



REFERENCES

1. For a more complete discussion of the derivation of Schroedinger's equation see Hanna, Melvin W. Quantum Mechanics in Chemistry, W. A. Benjamin, Inc., New York (1969) pp. 43-49 and Dewar, Michael, The Molecular Orbital Theory of Organic Chemistry. McGraw Hill Book Company, New York (1969) pp. 24-74.

2. This theorem states that "the expectation value of H for an arbitrary, well-behaved function Ψ is not less than the lowest eigenvalue E_0 of H ." see also Dewar, Michael, Molecular Orbital Theory of Organic Chemistry, p. 37. and Atkins, P. W., Molecular Quantum Mechanics, Clarendon Press, Vol. 1, Oxford (1970) pp. 214-215.

3. Pople, J.A., D. P. Santry, and G. A. Segal, J. Chem. Phys., 43. 5129 (1965).

4. Pople, J. A. and D. L. Beveridge, Approximate Molecular Orbital Theory McGraw Hill Book Company, New York (1970) pp. 11-12.

5. ibid., pp. 22-23.

6. ibid., pp. 37-41.

7. Pople, J. A., D. P. Santry, and G. A. Segal, J. Chem. Phys., 43. 5129 (1965).

8. Pople, J. A., and D. L. Beveridge, Approximate Molecular Orbital Theory pp. 62-79.

9. Pople, J. A., D. P. Santry and G. A. Segal, J. Chem. Phys., 43. 5129 (1965)

10. pp. 69-79.

11. Pople, J. A., and Beveridge, D. L., p. 72.

12. Santry, D. P., and G. A. Segal, J. Chem Phys., 47, 158 (1967).

13. Pople, J. A., D. L. Beveridge, and P. A. Dobosh, J. Chem. Phys, 47. 2076 (1967).

14. Pople, J. A., and G. A. Segal, J. Chem. Phys., 43, 5136 (1965).

15. Pople, J. A., D. L. Beveridge, and P. A. Dobosh, J. Chem. Phys., 47. 2026 (1967).

16. Pople, J. A., and M. S. Gordon, J. Am. Chem. Soc., 89. 4253 (1967).

17. Santry, D. P., and G. A. Segal, J. Chem. Phys., 47. 158 (1967).

18. Ronish, E. W. and S. Krimm, J. Chem. Phys., 56. 3175 (1972).

19. Jordan, F., and H. D. Sostman, J. Am. Chem. Soc., 94. 7898 (1972).

20. Imamura, A., H. Fujita, and C. Nagata, J. Am. Chem. Soc., 94. 6287 (1972).

21. Bruns, R. E., and W. B. Person, J. Chem. Phys., 57. 324 (1972).

22. Brownlee, R. T. C., and R. W. Taft, J. Am. Chem. Soc., 92. 7007 (1970).

23. Pople, J. A. and D. L. Beveridge, J. Chem. Phys., 49. 4725 (1968).

24. Baird, N. C., and M. J. S. Dear, J. Chem. Phys., 50. 1262 (1969).

25. Pople, J. A., D. P. Santry, and G. A. Segal, J. Chem. Phys., 43. 5129 (1965)

26. Sustmann, R., J. E. Williams, M. J. S. Dewar, L. C. Allen, P. von R. Schleyer, J. Am. Chem. Soc., 91. 5350 (1969).

27. Pople, J. A., and W. J. Heher, J. Am. Chem. Soc., 92. 2191 (1970).

28. ibid., 94. 6901 (1972).

29. R. Ditchfield, W. J. Hehre, and J. A. Pople, J. Chem. Phys. 54. 724 (1970).

30. ibid., 56. 2257 (1972).

31. Lathan, W. A., W. J. Hehre, and J. A. Pople, J. Am. Chem Soc. 93. 808 (1971).

32. Ditchfield, R., J. Del Bene, and J. A. Pople, J. Am. Chem. Soc., 94 4806 (1972).

33. Pople, J. A., D. P. Santry, and G. A. Segal, J. Chem. Phys., 43, 5129 (1965).

34. For more detail, the reader may consult Pople and Beveridge, Approximate Molecular Orbital Theory, pp. 80-83, 165.

35. Upchurch, Katherine, The Modification of a CNDO-INDO Computer Program and Its Application to the Study of the Effects of Fluorine and Chlorine on the Election Densities of Substituted Acetylenes and Ethylenes. (1972).

36. Pople, J. A., and D. L. Beveridge, Approximate Molecular Orbital Theory, p. 111.

37. Calculations on these molecules have also been previously done by K. Upchurch. See reference 35.

38. Morrison, R. T., and R. N. Boyd, Organic Chemistry, Allyn and Bacon, Inc., Boston (1969) pp. 175-202.

39. ibid, pp. 327-332.

40. ibid, pp. 360-365

96912YF TH
11/28/94 45190 4656

DATE DUE

GAYLORD PRINTED IN U.S.A.